CONTENTS

Section		Page	
	FOREWORD	iii	1/A5
1	INTRODUCTION	1	1/A7
2	METHOD DESCRIPTION	5	1/A11
	2.1 Equation Sets and the Quasi-Variational Technique	5	1/A11
	2.2 Geometric Treatment	8	1/A14
	2.3 Discretized Equations and Assembly	12	1/B4
	2.4 Nodal Analogs	16	1/B8
	2.5 Boundary Condition Treatment	22	1/B14
	2.6 Viscous Formulation and Stability	29	1/C7
	2.7 The GIM Code	32	1/C10
3	DEMONSTRATION PROBLEMS	34	1/C12
	3.1 Two-Dimensional Transonic Nozzle	34	1/C12
	3.2 Two-Dimensional Shock Capturing	37	1/D1
	3.3 Internal Ballistics Simulation	37	1/D1
	3.4 Three-Dimensional Supersonic Flow	41	1/D5
	3.5 Three-Dimensional Transonic Flow	45	1/D9
	3.6 Two-Dimensional Shear Flows	45	1/D9
	3.7 Three-Dimensional Shear Flows	53	1/E3
4	CONCLUDING REMARKS	66	1/F2
	4.1 Summary	66	1/F2
	4.2 Gas Dynamics	67	1/F3
	4.3 Aerodynamics	67	1/F3
5	REFERENCES	69	1/F5

NAS 1.26:3042

AUG 31 1978

NASA Contractor Report 3042

ORIGINAL
COMPLETED

Computation of Three-Dimensional Nozzle-Exhaust Flow Fields With the GIM Code

L. W. Spradley, P. G. Anderson,
and M. L. Pearson

PURCHASE ORDER L53289A
AUGUST 1978

NASA

Item 830-H-14 NAS1.26:3042

NASA Contractor Report 3042

Computation of Three-Dimensional Nozzle-Exhaust Flow Fields With the GIM Code

L. W. Spradley, P. G. Anderson,
and M. L. Pearson
Lockheed Missiles and Space Company, Inc.
Huntsville, Alabama

Prepared for
Langley Research Center
under Purchase Order L53289A

National Aeronautics
and Space Administration

**Scientific and Technical
Information Office**

1978

BLANK PAGE

FOREWORD

The purpose of this report is to present a description of the computational fluid dynamics capability of the General Interpolants Method (GIM) computer code. The work described in this report was performed by personnel of the Lockheed Missiles and Space Company, Huntsville Research and Engineering Center. It was supported, in part, by NASA-Langley Research Center under purchase order L53289A and L68133A. Inquiries concerning this report should be directed to James L. Hunt (NASA technical monitor, Hypersonic Aerodynamics Branch) or to:

Lawrence W. Spradley
Lockheed-Huntsville Research
& Engineering Center
4800 Bradford Drive
Huntsville, AL 35807
(205) 837-1800, extension 401

BLANK PAGE

1. INTRODUCTION

In recent years, extensive literature has appeared on the numerical integration of some form of the Navier-Stokes equations of fluid dynamics. The introduction of larger and faster computers has enabled investigators to experiment numerically with various approximate solution procedures for these complex equations. This development has proceeded along two different, but related, avenues: (1) mathematical solution of partial differential equations, and (2) engineering computation of physical phenomena. Although item one is not ignored, we are primarily concerned here with engineering analysis for obtaining useful results in practice. Acceptance of unsteady techniques in conjunction with larger and faster computers is changing the philosophy of computational mechanics since it is now possible to begin thinking of a general prediction tool for computing real world problems.

The system of equations considered here are various subsets of the Navier-Stokes equations plus an appropriate energy equation. Lagrangian coordinates or mixed Eulerian-Lagrangian frames are not considered. The equations are cast in unsteady form to pose the problem as initial valued in time. With appropriate spatial boundary values, the equations can be integrated forward in time in a parabolic/hyperbolic sense. The use of the unsteady formulation has a number of advantages: (1) the transient portion of a flow can be obtained as well as a steady state; (2) the asymptotic approach to a steady flow proceeds in much the same manner as nature; (3) subsonic, transonic and supersonic flows can be calculated with same methodology; (4) multi-dimensional flows can be handled more conveniently; (5) with a proper form of the differential equations, integration can automatically

proceed through shock waves or other discontinuities without undue complication; and (6) computation of viscous flows with time-fluctuating turbulence can be performed more naturally.

The art of computational mechanics is the construction of an appropriate numerical equivalent of the conservation equations since there is no systematic procedure which will yield the optimum approximation for a general system. There are at least two distinct ways to derive such approximation formulas. The first, or "grid" approach operates on the differential form of the equations by expanding the terms in a Taylor series and truncating it in some manner. The second approach, termed "cell" or "element" methods operate on the integral form of the equations and assumes a functional form for the variation of the dependent variables over a local finite volume. In either approach, the concepts of a "consistent" approximation, an "accurate" calculation procedure and a "stable" integration formula are all vital to obtaining solutions to the equations.

The classical works use finite difference techniques (Ref. 1) but recently the concept of the finite element, borrowed from structural analysis (Ref. 2), has become increasingly popular. Scientists with extensive backgrounds in finite difference solutions defended their approach against the finite element advocates. Likewise the finite element proponents claimed that the other side did not understand their new method. The resulting arguments have produced many papers and few solutions. An occasional attempt has even been made to relate the two approaches, but the schism still remained. It is the intention of this discussion to produce a consistent approach which clarifies the situation, and as it turns out, demonstrates that both approaches have merit and that each camp can learn a great deal from the other.

This report introduces a new methodology for constructing numerical analogs of the partial differential equations of continuum mechanics. A general formulation is provided which permits classical finite element methods and many of the finite difference methods to be derived directly. The approach,

termed the General Interpolants Method (GIM), is new in the sense that it can combine the best features of finite element and finite difference methods. The technique allows complex geometries to be handled in the finite element manner and operates on the integral form of the conservation laws. Solutions can be generated implicitly with the finite element analogs or by explicit finite difference analogs, which do not require a reduction of large systems of linear algebraic equations (no matrix inverse). A quasi-variational procedure is used to introduce boundary conditions into the method and to provide a natural assembly sequence for combining the element equations into the full domain equations.

The domain of interest is first discretized by appropriate subdivision into an assemblage of interconnected finite elements. A mesh generation is used in the GIM approach which incorporates general curvilinear coordinates, stretching transformations and bivariate blending to produce an automated mesh/element generation. Shape functions based on a set of generalized interpolants are then chosen to describe the behavior over each element. We then proceed, as in the Method of Weighted Residuals (Ref. 3) by multiplying the discretized equations by a set of weight functions and integrating over the volume of the element. A quasi-variational procedure is then used to construct the assembled system of equations from the element equations, and to introduce boundary conditions into the method. By choosing the weight functions equal to the shape functions, we reproduce via Galerkin the classical finite element nodal analogs. It is at this point that we introduce one of the important concepts of GIM: orthogonal weight/shape functions. By appropriately choosing the weight functions to be orthogonal to the shape functions, we can obtain explicit nodal analogs. Further, by a coice of arbitrary constants in the orthogonal weight functions, we can reproduce known finite difference nodal analogs, such as centered difference, upwind/downwind differences and the two-step MacCormack algorithm. As a result of this spatial discretization, we have reduced the partial differential equations to ordinary differential equations with "time" as the independent variable. Any forward marching algorithm such as Euler, Runge-Kutta or predictor-corrector can be used to advance the solution profiles in time.

In this report, we will use the unsteady equations of two-dimensional inviscid fluid flow to illustrate the GIM approach. Extension of this development

to three space dimensions and to viscous flow equations has been accomplished but we use the simpler problem to present the basic principles of GIM more tractably. The following points summarize GIM and its advantages over previous approaches.

- GIM is a higher order procedure for constructing computational analogs of the conservation laws of continuum mechanics.

- GIM is a total calculational procedure in that arbitrary geometries, expandable equation sets, different nodal analogs, general curvilinear coordinate systems and multi-constraint boundary values are handled.

- Non-orthogonal sets of weight/shape functions give rise to analogs which have generally been classified as finite element methods.

- Orthogonal sets of weight/shape functions give rise to explicit finite difference nodal analogs.

- A single analysis can be generated which selectively employs either of several finite element approaches or either of several finite difference techniques according to the wishes of the analyst.

- Finite Element discussions in the literature have mentioned quasi-variational methods, but none have consistently applied these ideas to boundary conditions as we have in GIM.

- The classical finite element analogs, derived via Galerkin, are unconditionally unstable for solution of the strong conservation form of the Euler equations for shock capturing. Several authors have presented formulations entitled Finite Element with automatic shock capturing, but no solutions are reported. We have found by numerical experiment, that the nodal analogs they derive are unstable. The only solutions presented make use of the theory of "weak solutions," which destroys the strong conservation form of the equations.

- The GIM approach allows the flexibility and generality of finite element techniques to be effectively married with proven successful finite difference techniques to produce a superior higher order methodology.

2. METHOD DESCRIPTION

2.1 EQUATION SETS AND THE QUASI-VARIATIONAL TECHNIQUE

The partial differential equations governing the two-dimensional flow of an inviscid, nonconducting ideal gas can be written in the following popular form:

$$\frac{\partial U}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + H = 0 \qquad (1)$$

where x, y are spatial Cartesian coordinates, t is time, U is a column vector of dependent flow variables, and E, F, H are nonlinear functions of U. For example,

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix} \qquad E = \begin{bmatrix} \rho u \\ P + \rho u^2 \\ \rho u v \\ (\rho E + P) u \end{bmatrix} \qquad F = \begin{bmatrix} \rho v \\ \rho u v \\ P + \rho v^2 \\ (\rho E + P) v \end{bmatrix} \qquad (2)$$

where ρ = density, u = velocity component in x, v = velocity component in y, E = total energy per unit mass and P = pressure.

This equation set, in conjunction with an equation of state $P = P(\rho, E)$ and appropriate boundary values, formally defines the type of problem used here to describe GIM.

This form of the equations is termed Strong Conservation Law or Divergence form. This form has been advantageous for finite difference proponents because of the superior characteristics in the neighborhood of discontinuities and steep gradients. It is also a convenient form for finite element analysis since the assembly procedure for collecting influence coefficients needs to be done only once.

GIM is developed in terms of equations of the form (1). Cartesian coordinates are used to write the equation sets and transformations are made numerically to relate a specific geometric region to this reference frame. Using Eq. (1) also allows the method to be formulated for multi-equation sets since the length of the column vectors is arbitrary. Thus additional equations can be added such as a third dimension, multi-specie reacting flow conservation laws and equations for multi-phase flows. Additional terms can also be added to this form to model viscous flows by including stress components, body forces, etc., to the E, F and H vectors.

It is natural to view Eq. (1) as a time-dependent system. With proper initial values and a realistic time, this form indeed represents the transient behavior of the flow. Additional ways of viewing Eq. (1) are also possible. With arbitrary initial conditions, the time t can be considered only as a relaxation coordinate such that Eq. (1) represents an asymptotic steady state form.

We may also view "t" as a spatial marching coordinate such that Eq. (1) represents a steady state hyperbolic system. Although a spatial marching variation of GIM is not precluded, we consider here only the unsteady or relaxation viewpoint of Eq. (1).

In many problems in mathematics and physics, a variational principle exists in which a stationary value of the principal function yields the governing equations. Perhaps the most complex, yet familiar, example is that of the maximization of entropy for closed systems leading to the equations governing chemical equilibrium.

There are few examples of field solutions obtainable through variational principles; the most notable in fluid mechanics is restricted to potential flow. There is no known principle that generates the primitive equations of motion for fluid mechanics.

In spite of the absence of a known principle we may, however, still take advantage of variational techniques. Let us assume a functional exists:

$$\mathcal{F}\left(\frac{\partial\rho}{\partial t}, \frac{\partial\rho u}{\partial t}, \frac{\partial\rho v}{\partial t}, \frac{\partial\rho E}{\partial t}\right)$$

whose extremum yields the primitive equation of motion. That is

$$\int_V \left[\frac{\partial\mathcal{F}}{\partial\left(\frac{\partial\rho}{\partial t}\right)}\,\delta\left(\frac{\partial\rho}{\partial t}\right) + \frac{\partial\mathcal{F}}{\partial\left(\frac{\partial\rho u}{\partial t}\right)}\,\delta\left(\frac{\partial\rho u}{\partial t}\right) + \frac{\partial\mathcal{F}}{\partial\left(\frac{\partial\rho v}{\partial t}\right)}\,\delta\left(\frac{\partial\rho v}{\partial t}\right) + \frac{\partial\mathcal{F}}{\partial\left(\frac{\partial\rho E}{\partial t}\right)}\,\delta\left(\frac{\partial\rho E}{\partial t}\right)\right] dV = 0$$

where

$$\frac{\partial\mathcal{F}}{\partial\left(\frac{\partial\rho}{\partial t}\right)} = \frac{\partial\rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y}$$

$$\frac{\partial\mathcal{F}}{\partial\left(\frac{\partial\rho u}{\partial t}\right)} = \frac{\partial\rho u}{\partial t} + \frac{\partial(p+\rho u^2)}{\partial x} + \frac{\partial(\rho uv)}{\partial y}$$

$$\frac{\partial\mathcal{F}}{\partial\left(\frac{\partial\rho v}{\partial t}\right)} = \frac{\partial\rho v}{\partial t} + \frac{\partial(\rho uv)}{\partial x} + \frac{\partial(p+\rho v^2)}{\partial y}$$

$$\frac{\partial\mathcal{F}}{\partial\left(\frac{\partial\rho E}{\partial t}\right)} = \frac{\partial\rho E}{\partial t} + \frac{\partial(\rho uE + up)}{\partial x} + \frac{\partial(\rho vE + vp)}{\partial y}$$

At the extremum, therefore, the coefficient of the variations must vanish, thereby satisfying the equations of motion. The above statements are described as a quasi-variational technique and where, in the absence of the variational principle, provide a rigorous academic point of departure for further developments. This type of idea is used in the GIM development to form an assembly procedure and to describe multi-constraint boundary values.

2.2 GEOMETRIC TREATMENT

The domain of interest is considered to be geometrically arbitrary in that any shape is represented as a bivariate blend of regular subdomains. Figure 2-1 shows an example of a full geometric domain in which flow is to be computed. This domain is subdivided into four regions as shown. In general this division is made such that analytical functions describe the shape of each side but it can, of course, be made by point specification and piecewise linear sides. Attention can then be focused on each region with global coordinates (x, y) and local region coordinates (ξ', η'). The regions are blended together at the junctions to provide the continuous full domain geometry.

We then proceed by considering each region separately as depicted in Fig. 2-2. Gordon and Hall (Ref. 5) show that the general relationship between physical space and local curvilinear coordinates is given by the following transformation:

$$R(\xi', \eta') = (1 - \xi')R(0, \eta') + \xi' R(1, \eta') + (1 - \eta')R(\xi', 0) + \eta' R(\xi', 1)$$

$$- (1 - \xi')(1 - \eta') R_1 - (1 - \xi')\eta' R_2 - \xi' \eta R_3 - \xi'(1 - \eta')R_4 \qquad (3)$$

where

$$R(\xi', \eta') = \begin{Vmatrix} x(\xi', \eta') \\ y(\xi', \eta') \end{Vmatrix}$$

is a two-component vector function and ξ', η' range from 0 to 1 along the contour of the region. The quantities R_1, R_2, R_3, R_4 are the x, y coordinates of the corners of the region and $R(0, \eta')$, $R(1, \eta')$, $R(\xi', 0)$, $R(\xi', 1)$ are geometric functions which describe the shape of the sides of the region (Fig. 2-2). To describe the geometry of a region thus requires input of four (x, y) coordinate pairs and four sets of functions to describe the sides. Figure 2-4 lists the library currently contained in the GIM code.

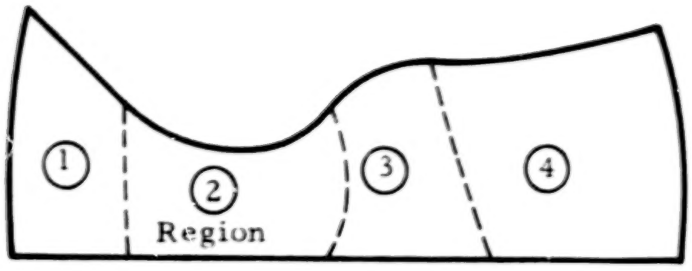


Fig. 2-1 - Flow Domain Example Subdivided into Regions

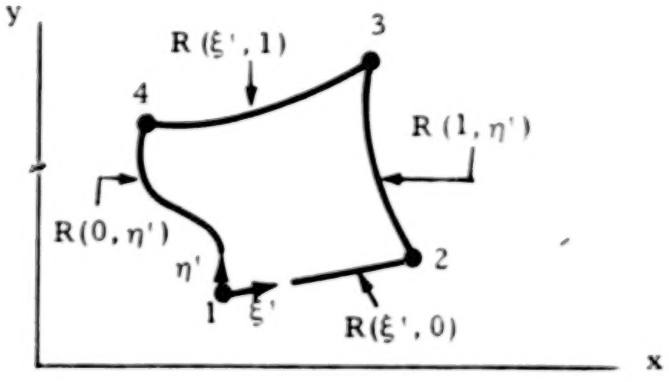


Fig. 2-2 - Global and Local Coordinate Systems for a Region

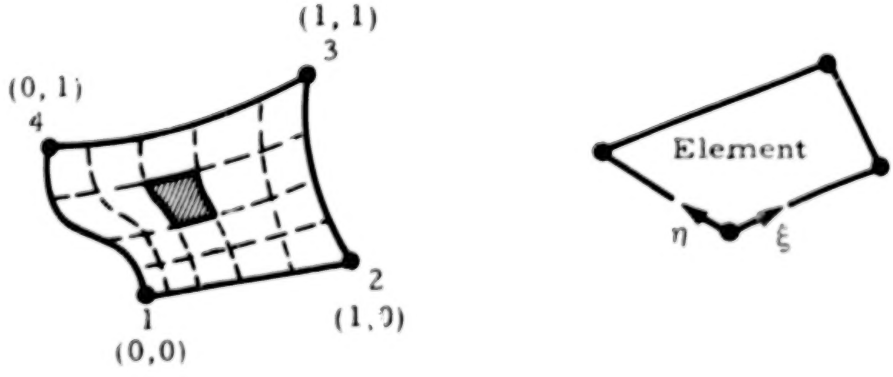


Fig. 2-3 - Element Breakdown for a Region

- HEXAHEDRAL BUILDING BLOCKS
  - EDGES (Combinations of up to five types)
    - Linear segment
    - Circular arc
    - Conic (elliptical, parabolic, hyperbolic) arc
    - Helical arc
    - Sinusoidal segment
  - SURFACES (Bounded by above edges)
    - Flat plate
    - Cylindrical surface
    - Edge of Revolution
- REGION OF INTEREST DESCRIBED BY UNLIMITED NUMBER OF HEXAHEDRAL BUILDING BLOCKS

Fig. 2-4 - Three-Dimensional Geometry Package

After determining the coordinates and geometric functions, we can then transform to (ξ', η') space and develop the equations in local coordinates. The next step is to subdivide each region into an assemblage of interconnected elements as depicted in Fig. 2-3. This is accomplished by simply specifying the number of node points in each direction. Each element is developed in its own local coordinates (ξ, η) in the same manner as the full domain was transformed into regions. All types of stretching transformations are possible with Eq. (3) so that irregular grid patterns can be used to enhance resolution. The elements are considered isoparametric in that the same functions will be used to approximate the flow variables and the element geometry. For example, if linear interpolants are used for velocity, pressure, etc., then a typical element shown in Fig. 2-3 will have linear sides.

Within each element, any function, Q, will be described by a set of interpolants, S, such that

$$Q(\xi, \eta) = S_i(\xi, \eta)\, Q(\xi_i, \eta_i) \tag{4}$$

$$= S_i Q_i \quad \text{summed } i = 1, k$$

where Q_i is the function at the node points of the element and k is the number of nodes on the element. The derivatives of any function Q in (ξ, η) space is then

$$\frac{\partial Q}{\partial \xi} = \frac{\partial S_i}{\partial \xi} Q_i \qquad \frac{\partial Q}{\partial \eta} = \frac{\partial S_i}{\partial \eta} Q_i \quad \text{summed } i = 1, k \tag{5}$$

since Q_i is a point function.

In order to relate such derivatives back to physical (x, y) space, we use the following relation from the calculus.

$$\begin{Bmatrix} \frac{\partial Q}{\partial x} \\ \\ \frac{\partial Q}{\partial y} \end{Bmatrix} = [J]^{-1} \begin{Bmatrix} \frac{\partial Q}{\partial \xi} \\ \\ \frac{\partial Q}{\partial \eta} \end{Bmatrix} \tag{6}$$

where $[J]$ is the Jacobian matrix of the transformation, defined as

$$[J] = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \tag{7}$$

This is easily obtained by differentiating Eq. (3). The integration over an element also requires the area transformation

$$dx\,dy = \Delta_J\, d\xi\, d\eta \tag{8}$$

where Δ_J is the determinant of the Jacobian matrix $[J]$.

We now have all of the geometric information to develop the discretized conservation laws for an element.

2.3 DISCRETIZED EQUATIONS AND ASSEMBLY

With the coordinates and geometric shape functions now defined for each element, we proceed to discretize the differential equations for an individual element. The procedure follows that of the Method of Weighted Residuals (Ref. 3) in that we require the weighted integral of the differential equations to be zero over the domain, R.

$$I = \int_R W \left(\frac{\partial U}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + H \right) dx\,dy = \int_R W \mathcal{D}\, dA = 0 \tag{9}$$

where W is an arbitrary weighting function. Equation (9) must be satisfied over the domain R and can be written as a summation of integrals over each element

$$I = \sum_{e} \int_{R_e} W\mathcal{D}\, dA \tag{10}$$

Equation (10) then represents the point of departure for obtaining the discrete conservation equations. The integrals over each element R_e are developed by applying the general interpolant shape function, S (Eq. (4)) to the differential equation (Eq. (1)). For simplicity we will use the same shape functions for each flow variable although this is not a restriction on the approach.

The dependent variables are thus assumed to be given by the following relations over an element

$$\begin{aligned} \dot{U} &= S_j \dot{U}_j \qquad E = S_j E_j \\ F &= S_j F_j \qquad H = S_j H_j \end{aligned} \tag{11}$$

where summation $j = 1, k$ is implied, k is the number of nodes on an element ($k = 4$ in Fig. 2-3), and $\dot{U} = \frac{\partial U}{\partial t}$.

For any weight function W_i, we put Eq. (11) into Eq. (10) for a single element to yield

$$I_e = \int_y \int_x W_i \left[S_j \dot{U}_j + \frac{\partial S_j}{\partial x} E_j + \frac{\partial S_j}{\partial y} F_j + S_j H_j \right] dx\, dy \tag{12}$$

Now we note that $\dot{U}_j$ E_j, F_j, H_j are point functions dependent only on time, so that we evaluate the integrals in terms of known functions, W_i, S_j, $\frac{\partial S_j}{\partial x}$, $\frac{\partial S_j}{\partial y}$. Let the following element matrices be defined.

$$[A_{ij}^e] = \iint_y \iint_x W_i S_j \, dx \, dy$$

$$[B_{ij}^e] = \iint_y \iint_x W_i \frac{\partial S_j}{\partial x} \, dx \, dy \qquad (13)$$

$$[C_{ij}^e] = \iint_y \iint_x W_i \frac{\partial S_j}{\partial y} \, dx \, dy$$

Equation (12) then becomes

$$I_e = [A_{ij}^e](\dot{U}_j + H_j) + [B_{ij}^e] E_j + [C_{ij}^e] F_j \qquad (14)$$

(summed $j = 1, k$)

Equation (14) is a set of k ordinary differential equations for an element e. The integrals (Eq. (13)) can best be evaluated by invoking the (ξ, η) transformations (Eqs. (6)–(8)) to yield

$$[A_{ij}^e] = \int_0^1 \int_0^1 W_i S_j \Delta_J \, d\xi \, d\eta$$

$$[B_{ij}^e] = \int_0^1 \int_0^1 W_i \left(\frac{\partial S_j}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial S_j}{\partial \eta} \frac{\partial y}{\partial \xi} \right) d\xi \, d\eta \qquad (15)$$

$$[C_{ij}^e] = \int_0^1 \int_0^1 W_i \left(\frac{\partial S_j}{\partial \eta} \frac{\partial x}{\partial \xi} - \frac{\partial S_j}{\partial \xi} \frac{\partial x}{\partial \eta} \right) d\xi \, d\eta$$

The discretized element equations are thus given by Eq. (14) with the coefficient matrices given by the geometric integrals Eq. (15). Note that in Eq. (14) the j index is summed and the i index counts the node number of the equation.

Since the choice of W is arbitrary, the integral must be stationary for variations in W. The so-called quasi-variational approach determines the conditions under which the integral is stationary. That is to say that if $W' = W + \delta W$ then

$$\delta I = \sum_{e=1}^{N} \int_{R_e} \delta W \, \mathcal{D} \, dxdy = 0$$

Now let

$$W = W_j (\xi, \eta) \, \dot{U}_j \qquad \text{or}$$

$$\delta W = W_j \, \delta \dot{U}_j$$

so that

$$\delta I = \sum_{e=1}^{N} \left\{ \int_{R_e} W_j \, \mathcal{D} \, dxdy \, \delta \dot{U}_j \right\} = 0$$

The for δI to be stationary, the sum of the coefficients of $\delta \dot{U}_j$ must be zero. This provides a rationale for the classical assembly procedure of MWR derived finite element schemes.

In mathematics, this can be written as

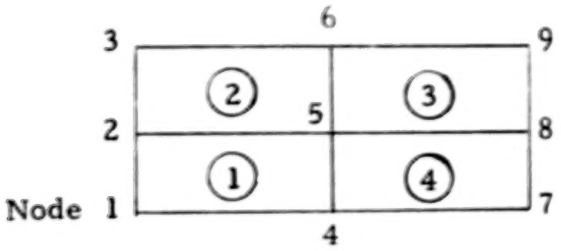
$$\left[A_{mn}\right] = \sum_{e} \delta_{mi} \left[A^{e}_{ij}\right] \delta_{nj} \tag{16}$$

where

$$\delta_{mi} = \begin{cases} 1 & \text{if node i of element e coincides with node m of the region R} \\ 0 & \text{otherwise} \end{cases}$$

and similarly for the B, C matrices. The rationale for doing this is directly derivable from the quasi-variational procedure.

A simple example of the assembly procedure is given for the following four element domain.



Node 1 of the full domain contains only element matrices from element (1) while node 2 of the full domain contains a sum of element matrix components from elements (1) and (2). Node 5 of the full domain contains a sum of matrix components from all four adjacent elements.

With the nomenclature of Eq. (15), the full domain equations are

$$[A_{mn}]\ (\dot{U}_n + H_n) + [B_{mn}]\ E_n + [C_{mn}]\ F_n = 0 \qquad (17)$$

Note again that in Eq. (17), the n index is summed over the number of nodes and the m index counts equations for each node. In addition, we must recall that $\dot{U}, E, F, H$ are column vectors themselves. For two-dimensional flow, Eq. (17) then represents 4k equations for k nodes in a domain, R. The spatial analogs and geometric transformations are all contained in the coefficient matrices. Thus, Eq. (17) is a system of ordinary differential equations which, with suitable initial data, can be integrated as an initial value problem.

2.4 NODAL ANALOGS

At this point, Eq. (17) is general in that the geometric functions are arbitrary and the flow variable shape functions and weight functions are arbitrary. To this point the development is similar to the general Method of Weighted Residuals (MWR) (Ref. 3) with the geometric transformations built into the integrals. Readers familiar with MWR will realize that closed

form evaluation of the integrals (Eq. (15)) is impossible in general due to presence of the Jacobian inverse components for an arbitrary transformation. In the GIM code we evaluate all element integrals using Gaussian quadrature to maintain generality of geometry.

If we choose the weight functions W to be identical to the shape functions S, then the Eqs. (17) represent the Galerkin approach to finite element nodal analogs. Inspection of these equations will reveal that, in general, the spatial nodal analogs are implicit in that the unsteady derivatives $\dot{U}$ are summed over all connecting node points. In this work we will refer these "finite element" equations as implicit nodal analogs. Any other classical choice of weight functions are possible at this point in the analysis, such as collocation, least squares, etc.

We now turn to one of the major developments of the GIM analysis; the generation of explicit nodal analogs from the MWR point of departure. In this work we will refer to explicit analogs as "finite difference" equations not to preclude the fact that implicit finite difference analogs are also possible. To consider explicit expressions we turn to Eq. (17) and note that the $[A_{mn}]$ matrix is the key to rendering the algorithms explicit. If the $[A^e_{ij}]$ element matrices were made diagonal then the $[A_{mn}]$ full domain matrix will be diagonal. This would reduce Eq. (17) to an explicit expression for the unsteady derivatives, $\dot{U}$.

To accomplish this goal, we choose the weight functions W to be <u>orthogonal</u> to the shape functions, S.

$$[A^e_{ij}] = \int_0^1 \int_0^1 W_j S_i \Delta_J \, d\xi \, d\eta = 0 \quad i \neq j \tag{18}$$

Equations (17) then become

$$\dot{U}_m = - \frac{1}{a_{mm}} \left\{ [B_{mn}] E_n + [C_{mn}] F_n \right\} - H_m$$

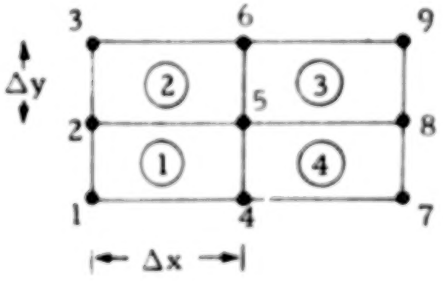
where a_{mm} are the diagonal (non-zero) components of $[A_{mn}]$.

This choice of weight functions does not reproduce the "classical" finite element form but rather an explicit nodal analog which we may view as a finite difference type of analog. The $[B_{mn}]$ and $[C_{mn}]$ matrices now resemble "derivative takers" in that $\frac{\partial E}{\partial x}$ is approximated by

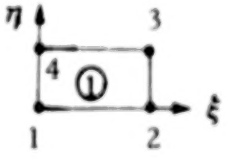
$$\left. \frac{\partial E}{\partial x} \right|_m \approx \frac{1}{a_{mm}} \sum_{n=1}^{N} B_{mn} E_n$$

The specific type of finite difference analog which is produced is a function of the element shape functions, S, and the orthogonal weight functions, W. To clarify this situation and to show that such expressions do yield proven finite difference formulas, the following simple example is given.

Consider the following four-element, nine-node rectangular region:



Full Domain
Numbering System



Individual Element
Numbering System

The simplest shape functions are the linear interpolants

$$S = \begin{Bmatrix} (1-\xi)(1-\eta) \\ \xi\,(1-\eta) \\ \xi\,\eta \\ (1-\xi)\eta \end{Bmatrix} \tag{19}$$

Using these shape functions in the orthogonality condition, Eq. (18), we find the following weight functions.

$$W = \frac{1}{\Delta x\,\Delta y} \begin{Bmatrix} \alpha_1 (2/3-\xi)(2/3-\eta) \\ \alpha_2 (1/3-\xi)(2/3-\eta) \\ \alpha_3 (1/3-\xi)(1/3-\eta) \\ \alpha_4 (2/3-\xi)(1/3-\eta) \end{Bmatrix} \tag{20}$$

where the α_i are arbitrary constants.

Putting these S and W functions into the integral coefficients, Eq. (15), and carrying out the integration analytically, we get the following element matrices.

$$[A^e_{ij}] = \frac{1}{36} \begin{bmatrix} \alpha_1 & 0 & 0 & 0 \\ 0 & -\alpha_2 & 0 & 0 \\ 0 & 0 & \alpha_3 & 0 \\ 0 & 0 & 0 & -\alpha_4 \end{bmatrix} \tag{21}$$

$$[B^e_{ij}] = \frac{1}{36\Delta x}\begin{bmatrix} -\alpha_1 & \alpha_1 & 0 & 0 \\ \alpha_2 & -\alpha_2 & 0 & 0 \\ 0 & 0 & \alpha_3 & -\alpha_3 \\ 0 & 0 & -\alpha_4 & \alpha_4 \end{bmatrix} \tag{21}$$

$$[C^e_{ij}] = \frac{1}{36\Delta y}\begin{bmatrix} -\alpha_1 & 0 & 0 & \alpha_1 \\ 0 & \alpha_2 & -\alpha_2 & 0 \\ 0 & -\alpha_3 & \alpha_3 & 0 \\ \alpha_4 & 0 & 0 & -\alpha_4 \end{bmatrix}$$

Now we can assemble the coefficient matrices for the center node (Node 5) and examine the resulting analog. Applying the assembly technique, Eq. (16), the resulting analog for node 5 is the following:

$$(\alpha_1 - \alpha_2 + \alpha_3 - \alpha_4)\,[\dot{U}_5 + H_5]$$

$$+ \left[\frac{(\alpha_1 - \alpha_4)\,E_8 + (\alpha_3 + \alpha_4 - \alpha_1 - \alpha_2)\,E_5 + (\alpha_2 - \alpha_3)\,E_2}{\Delta x}\right]$$

$$+ \left[\frac{(\alpha_1 - \alpha_2)\,F_6 + (-\alpha_1 + \alpha_2 + \alpha_3 - \alpha_4)\,F_5 + (\alpha_4 - \alpha_3)\,F_4}{\Delta y}\right]$$

$$= 0 \tag{22}$$

Equation (22) is the general finite difference type of analog resulting from the orthogonality condition. Since the α_i are arbitrary constants we can choose them to give a choice of actual nodal analog. Let

$$\alpha_1 = \alpha_3 = 1 \qquad \alpha_2 = \alpha_4 = -1$$

Equation (22) then becomes

$$\dot{U}_5 + H_5 + \frac{E_8 - E_2}{2\Delta x} + \frac{F_6 - F_4}{2\Delta y} = 0 \qquad (23)$$

Equation (23) is readily recognized as a space centered scheme which is unconditionally unstable for an explicit single step time integration. The important point is, however, that it is a finite difference scheme. The conclusion is that finite difference schemes (explicit nodal analogs) arise from orthogonal weight and shape function sets while classical finite element schemes (implicit nodal analogs) result from non-orthogonal choices.

By selecting the α_i in various manners other finite difference schemes may be generated. For example let

$$\alpha_1 = 1 + \mu;\ \alpha_3 = 1 - \mu;\ \alpha_2 = \alpha_4 = -1$$

then

$$\dot{U}_5 + \frac{E_8 - E_2}{2\Delta x} + \frac{F_6 - F_4}{2\Delta y} + \frac{\mu\Delta x}{4}\frac{(E_8 - 2E_5 + E_2)}{\Delta x^2} + \frac{\mu\Delta y}{4}\frac{(F_6 - 2F_5 + F_4)}{\Delta y^2} = 0 \qquad (24)$$

which is just Eq. (23) with some psuedo-viscous terms added.

A more pertinent example is a two-step method in which we choose for the first step $\alpha_2 = \alpha_3 = \alpha_4 = 0;\ \alpha_1 = 1$ while for the second step $\alpha_1 = \alpha_2 = \alpha_4 = 0;\ \alpha_3 = 1$. We then have for the first (prediction step)

$$\dot{U}_5 + \frac{E_8 - E_5}{\Delta x} + \frac{F_6 - F_5}{\Delta y} = 0 \qquad (25a)$$

while for the second step using provisional values generated above

$$\dot{U}_5 + \frac{E_5 - E_2}{\Delta x} + \frac{F_5 - F_4}{\Delta y} = 0 \qquad (25b)$$

Equations (25) are the MacCormack technique (Ref. 6) which has become very popular in recent years. Further it should be pointed out that different assemblies may be used for different equations yielding combination schemes of forward, backward and centered which have been successfully used in problems involving meteorological phenomena and convection analyses.

Some (perhaps not all) finite difference techniques can therefore be generated from the finite element point of departure. These schemes are particularly attractive for large scale problems where the implicit nature of the non-orthogonal weighting functions create an insurmountable obstacle to the successful pursuit of the solution. Another important feature is that a great deal is known about stability characteristics of some of the explicit schemes. An observation which is felt to be of paramount importance is that it is not necessary to make the choice of schemes a priori. Rather a general computer program can be written in which the user chooses his assembly techniques upon input rather than at inception of the coding effort. Thus, a nonorthogonal choice may be made if the number of unknowns is small while an orthogonal choice can be made if the number of unknowns is large. This is a tendency then toward a very general finite difference/finite element fluid mechanics computer code.

2.5 BOUNDARY CONDITION TREATMENT

The boundary condition treatment in the GIM code is based on a quasi-variational technique. The discussion here will briefly outline the procedure.

In this discussion we will assume that any number of constraints are acting simultaneously. We generalize the constraint statement to read:

$$\Gamma_i = \alpha_{ik} \dot{U}_i + \beta_i = 0; \quad k = 1, \ldots, K; \quad i = 1, \ldots, I \tag{26}$$

There are then K-I independent motions. Rewritting Eq. (26) yields

$$\sum_{k=1}^{K-I} \alpha_{ik} \dot{U}_k + \sum_{\ell=K-I+1}^{K} \alpha_{i\ell} \dot{U}_\ell + \beta_i = 0 \tag{27}$$

where the ℓ motions are considered dependent while the k subscript motions are assumed independent. In matrix form

$$[L]\{\dot{U}_k\} + [\mathcal{L}]\{\dot{U}_\ell\} + \{\beta\} = 0 \tag{28}$$

where $[\mathcal{L}]$ is a square (I x I) matrix. The dependent motions are found explicitly from:

$$\{\dot{U}_\ell\} = -[\mathcal{L}]^{-1}\left\{[L]\{\dot{U}_k\} + \{\beta\}\right\} \tag{29}$$

The variational statement is

$$\{\delta\dot{U}_\ell\} = -[\mathcal{L}]^{-1}\ [L]\{\delta\dot{U}_k\} \tag{30}$$

After much algebraic manipulation we get the following expression for the independent motions.

$$\hat{\dot{U}}_k = [D^{-1}]\left\{\dot{U}_k - [\tilde{L}][\mathcal{L}^{-1}]\left([\mathcal{L}^{-1}]\beta + \dot{U}_k\right)\right\} \tag{31}$$

where

$$[D] = [I] + [\tilde{L}][\mathcal{L}^{-1}][\mathcal{L}^{-1}][L]$$

the tilde ($\sim$) indicates transpose and the hat ($\wedge$) indicates new values now corrected for boundary constraints. Equation (31) is used to compute the independent motions.

As an example, consider the case of inviscid wall tangency

$$\dot{q} \cdot \vec{N} = n_x\dot{u} + n_y\dot{v} + n_z\dot{w} = 0 \tag{32}$$

where $\vec{N}$ is the unit normal to the wall.

The constraint coefficients are then

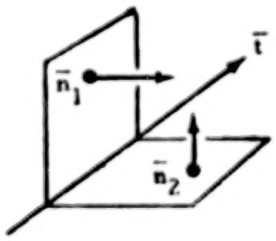
$$[L] = [n_x\ n_y];\ [\mathcal{L}] = n_z;\ \beta = 0$$

From Eqs. (29) and (31) we get the following equations for the following equations for the unsteady velocity derivatives.

$$\hat{\dot{U}}_i^k = \dot{U}_i^k - N_i^k \sum_{j=1}^{3} N_i^j \dot{U}_i^j \tag{33}$$

where the N_i^j are unit normal components at the wall point i (determined from geometry). Inspection will show that this maintains flow tangency if the initial t = 0 data are tangent to the wall.

Consider now the multiple constraint situation (internal corner):



where the flow is constrained to be tangent to two surfaces hence must flow parallel to the corner. The equations of constraint are:

$$\begin{aligned} \dot{u}n_{1x} + \dot{v}n_{1y} + \dot{w}n_{1z} &= 0 \\ \dot{u}n_{2x} + \dot{v}n_{2y} + \dot{w}n_{2z} &= 0 \end{aligned} \tag{34}$$

so that

$$[L] = \begin{Bmatrix} n_{1x} \\ n_{2x} \end{Bmatrix};\ [\mathcal{L}] = \begin{bmatrix} n_{1y} & n_{1z} \\ n_{2y} & n_{2z} \end{bmatrix};\ \{\beta\} = 0 \tag{35}$$

Again Eqs. (29) and (31) give the appropriate expression for corner flow:

$$\hat{\dot{U}}_i^k = T_i^k \sum_{j=1}^{3} T_i^j \dot{U}_i^j \tag{36}$$

where i is the nodal point nunber and k = 1, 2, 3 for each velocity component. The T_i^k are the unit tangent vector components for the corner (determined from the geometry).

The following are the type of nodal point conditions which the GIM code can treat.

Type	Condition
0	Known or fixed conditions
1	Nodes having the same value as another
2	Stagnation point
3	Internal corner flow
4	Wall tangency (slip)
5	Not currently used
6	Not currently used
7	Not currently used
8	Wall point determined from one-sided differences
9	Interior flow points.

The treatment of each type is in terms of the unsteady derivatives, $\dot{U}$, and are given by the following:

Type 0

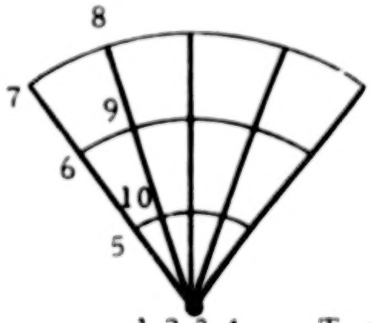
This is a boundary node at which all flow variables are specified such as an upstream inflow to a duct. Mathematically, this condition is

$$\dot{U}_i = 0 \quad \text{for all nodes i to be specified}$$

This requires, of course, that the U vector itself be properly initialized to the inflow conditions.

Type 1

This is a boundary condition for points which are common to more than one element such as:



Nodes 1, 2, 3 and 4 are actually the same point but four values may be determined from flow in each of four elements. Mathematically, this is treated as follows:

$$\dot{U}_i = \frac{1}{N} \sum_{j=1}^{N} \dot{U}_j \qquad i = 1, N \tag{37}$$

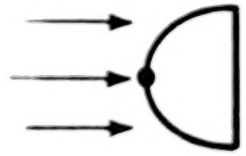
Type 2

A type 2 node is a stagnation point for which

$$U_i = 0 \quad \text{for components of } U_i \text{ which are flow velocities}$$

Other components of U such as density and energy are integrated using the nodal analogs.

Type 2



Type 3

Flow in a corner is forced to flow along the corner by imposing conditions on the components of $\dot{U}$ which are velocity variables

$$\hat{\dot{U}}_i^k = T_i^k \sum_{j=1}^{3} T_i^j \dot{U}_i^j \tag{38}$$

where i is the nodal point number and k = 1, 2, 3 for each velocity component. The T_i^k are the tangent vector components for the corner (determined from the geometry and the hat (^) indicates corrected values).

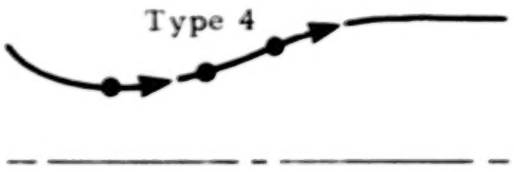
Type 3

Type 4

Inviscid flow tangency or free slip conditions are specified by imposing the foliowing conditions on the velocity component of $\dot{U}$:

$$\hat{\dot{U}}_i^k = \dot{U}_i^k - N_i^k \sum_{j=1}^{3} N_i^j \dot{U}_i^j \tag{39}$$

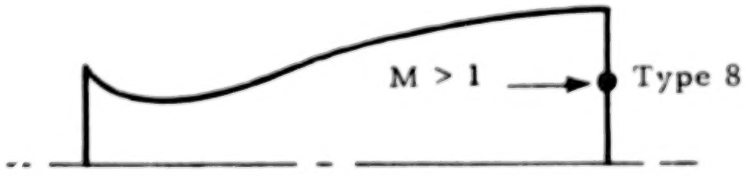
where the N_i^k are the unit normal components at the wall point i (determined from geometry and the hat ( ) indicates corrected values).



This approach maintains tangency if the initial t = 0 data are tangent to the wall.

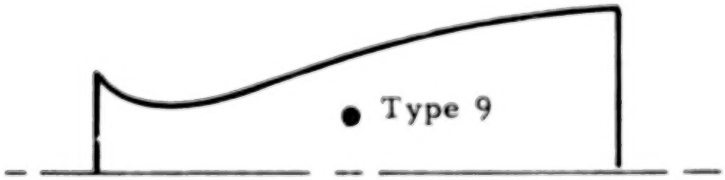
Type 8

Type 8 boundary nodes are those in which flow is allowed to exit the domain at an unspecified rate. Such a condition is the downstream wall in a supersonic flow where the wall values are determined from upstream (backward) differences. Wall points of this type are computed from one-sided differences determined by a proper choice of the α_i to yield one-sided nodal analogs at the boundaries.



Type 9

These are interior points in the flow domain which are to be unrestricted and determined entirely from the nodal analog of the differential equations.



2.6 VISCOUS FORMULATION AND STABILITY

The full set of differential equations solved by the GIM code is shown in Fig. 2-5. These will be referred to as the Navier-Stokes equations, although they include continuity and energy conservation as well as momentum. For inviscid flow calculations without shock waves, the σ and τ terms are set to zero to produce the Euler set. Figure 2-6 lists the viscous terms with coefficients, μ, λ, k for the first and second coefficients of viscosity and the thermal conductivity, respectively. These are specified by the user as "laminar" constants.

For reasons of numerical stability and capture of strong shock waves, additional components of the diffusion coefficients are added automatically by the GIM code. Figure 2-7 is a list of currently used Numerical Diffusion Cancellation (NDC) coefficients. These are added to the real diffusion coefficients. The purpose of these NDC coefficients is to cancel low order truncation error terms which arise in the numerics. A forthcoming paper by Spradley will present the basic principles of the NDC technique in more detail.

The differential equations are solved in strong conservation or divergence law form. The solution is started at some time t where the entire flow field mesh is specified. The unsteady or relaxation of the equations is then done using the user-specified nodal analogs. At this time, the pressure is computed from the ideal gas law for a single component gas. The code is being modified to include multi-gas capability and other additions.

Stability of the unsteady solution is maintained through addition of the viscous cancellation terms. The time step is limited by the classical inviscid CFL criterion or the diffusion time criteria, whichever is smaller. Adding too much artificial viscosity smears the solution and also reduces the allowable time step for stability. The GIM code adds just enough viscosity to balance the low order truncation errors.

$$\frac{\partial U}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + \frac{\partial G}{\partial z} = 0$$

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{bmatrix}$$

$$E = \begin{bmatrix} \rho u - \sigma_x \\ \rho u^2 + P - \tau_{xx} \\ \rho uv - \tau_{xy} \\ \rho uw - \tau_{xz} \\ (\rho E + P)u - u\tau_{xx} - v\tau_{xy} - w\tau_{xz} - q_x \end{bmatrix}$$

$$F = \begin{bmatrix} \rho v - \sigma_y \\ \rho vu - \tau_{xy} \\ \rho v^2 + P - \tau_{yy} \\ \rho vw - \tau_{yz} \\ (\rho E + P)v - u\tau_{xy} - v\tau_{yy} - w\tau_{yz} - q_y \end{bmatrix}$$

$$G = \begin{bmatrix} \rho w - \sigma_z \\ \rho wu - \tau_{xz} \\ \rho wv - \tau_{yz} \\ \rho w^2 + P - \tau_{zz} \\ (\rho E + P)w - u\tau_{xz} - v\tau_{yz} - w\tau_{zz} - q_z \end{bmatrix}$$

Fig. 2-5 - Three-Dimensional Cartesian Navier-Stokes Conservation Laws

$$\tau_{xx} = 2\mu \frac{\partial u}{\partial x} + \lambda \nabla \cdot \vec{V}$$

$$\tau_{yy} = 2\mu \frac{\partial v}{\partial y} + \lambda \nabla \cdot \vec{V}$$

$$\tau_{zz} = 2\mu \frac{\partial w}{\partial z} + \lambda \nabla \cdot \vec{V}$$

$$\tau_{xy} = \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)$$

$$\tau_{xz} = \mu \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)$$

$$\tau_{yz} = \mu \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right)$$

$$q_x = k \frac{\partial T}{\partial x}$$

$$q_y = k \frac{\partial T}{\partial y}$$

$$q_z = k \frac{\partial T}{\partial z}$$

Fig. 2-6 - Viscous Terms in GIM Code

$$\sigma_x = \epsilon_1 (u^2 + c^2) \frac{\partial \rho}{\partial x} \Delta t$$

$$\sigma_y = \epsilon_1 (v^2 + c^2) \frac{\partial \rho}{\partial y} \Delta t$$

$$\sigma_z = \epsilon_1 (w^2 + c^2) \frac{\partial \rho}{\partial y} \Delta t$$

$$\mu_d = \epsilon_2 \rho \frac{\Delta t}{2} (u^2 + v^2 + w^2)$$

$$\mu = \mu_{real} + \mu_d$$

$$\lambda = c_1 \mu \qquad c_1, c_2 = O(1)$$

$$k = c_2 \mu \qquad \epsilon_1, \epsilon_2 = O(1)$$

Fig. 2-7 - Example of the Stability Coefficients in GIM Code

2.7 THE GIM CODE

The GIM code is divided into four modules: (1) mesh generation; (2) nodal analog assembly; (3) unsteady integration; and (4) data display. The mesh generation module accepts boundary geometry data, curve or line formula flags, and number of cuts in each coordinate direction. A set of general curvilinear coordinate maps is then used to subdivide each region into finite elements. Each region which is input is processed and then blended. The output is a set of coordinates for each element along with the element coefficient matrices. The nodal analog assembly module takes the mesh data from a stored external file and performs, via quasi-variational procedure, the assembly of the element equations into the full domain equations. At this point, the dynamic storage allocation is set up so that the unsteady integration module can integrate with virtually unlimited problem size.

The unsteady integration module performs the actual computation of the flow by employing the boundary conditions selected by the user. The nodal analog at this point is arbitrary and any one of a number of schemes can be selected depending on the problem being analyzed. The solution is marched forward in time for a specified number of steps or until a steady state is reached. The data display module reads the solution profiles from external storage (drum, tape) and prints, plots and maps the flow parameters. Figure 2-8 is a block diagram illustrating the modular construction of the GIM code. Figure 2-9 is a summary of the computer utilization of the code.

A study is currently underway to determine the feasibility and practicality of implementing the GIM code on a CDC STAR computer system. The pipelining features of such a machine appear to be well suited to the GIM formulation since the numerical analogs are written as vector products and matrix-vector products. The current STAR features also appear to be best for explicit methods rather than implicit schemes. The feasibility study is being supported by the NASA-Langley Research Center.

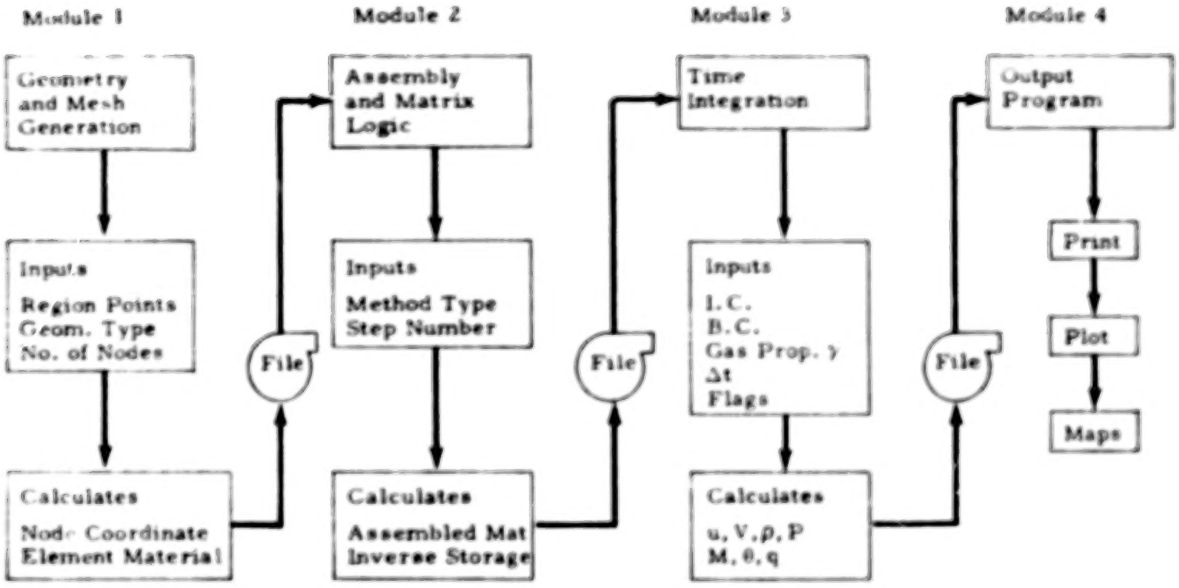


Fig. 2-8 - GIM Computer Code Structure

- OPERATIONAL ON CDC 7600, UNIVAC 1100 SERIES
- DYNAMIC STORAGE ALLOCATION
- 7600 CORE REQUIREMENTS
  - $100K_8$ SCM
  - $50K_8$ LCM (Nominal)
  - 9 External Files
- EXECUTION TIMES (7600)
  - Inviscid 4×10^{-4} sec/node - Iter
  - Viscous 6×10^{-4} sec/node - Iter
  - Example - 5000 nodes, 300 Iter
  - Inviscid 10 min CP
  - Viscous 15 min CP

Fig. 2-9 - Computer Utilization Summary

3. DEMONSTRATION PROBLEMS

Seven sample cases were selected for presentation in this technical brief because these cases illustrate the utility of the methods and also reveal some interesting results for relatively complex flows. The equations are the same for all of the problems, but the geometry and boundary conditions are different. The equations are the Euler equations plus certain viscous terms for an ideal gas written in strong conservative or divergence form. This form permits the use of "shock capturing" algorithms rather than "shock fitting" algorithms. The MacCormack scheme is used in the computations for illustrative purposes.

3.1 TWO-DIMENSIONAL TRANSONIC NOZZLE

This case consists of the single phase two-dimensional transonic flow in a converging-diverging nozzle with a finite area inlet and a 15 deg exit using air as the fluid. The upstream boundary condition consists of specified Mach number, density and pressure along a grid arc. Flow tangency is prescribed along the centerline and upper wall. The flow will thus be subsonic upstream of the throat, then expand to supersonic conditions in the divergent section of the region. Figure 3-1 shows the problem geometry and mesh arrangement used in the GIM computation. Quadrilateral elements are used and the method selected is the two-step MacCormack scheme. Figure 3-2 shows the steady state flow vectors on the physical mesh. The length of the arrow gives relative magnitude of the velocity. Figure 3-3 is a computer generated plot of the steady state Mach contours in the nozzle. These values have been compared to measured profiles (Ref. 7) for this configuration for the sonic line.

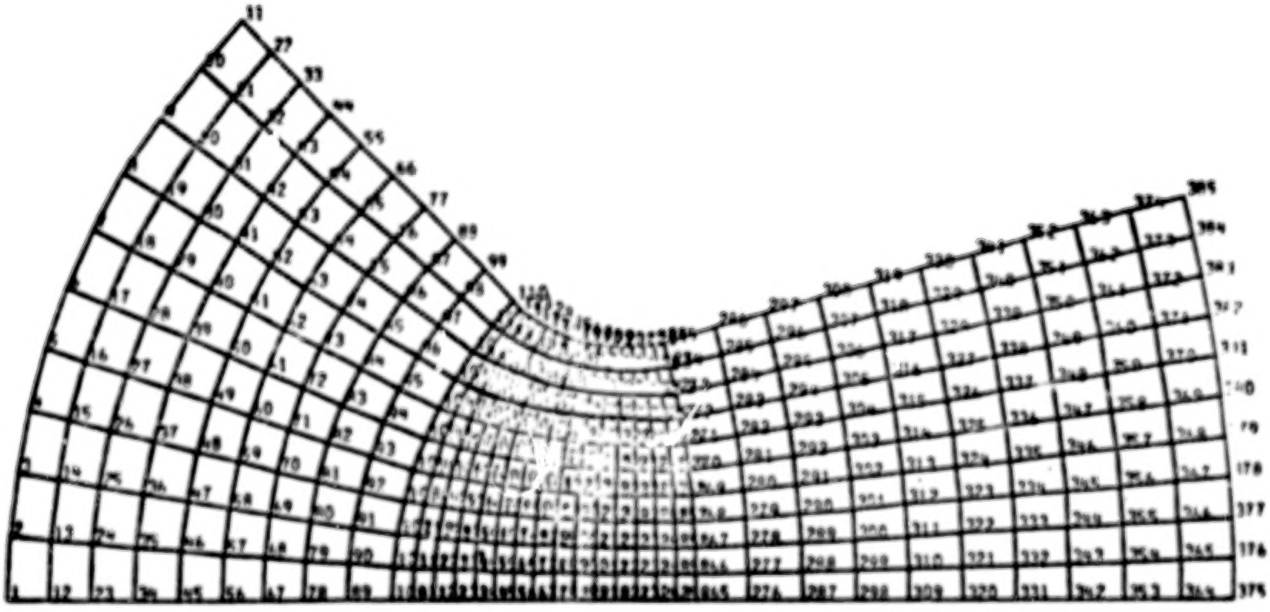


Fig. 3-1 - Geometry for Transonic Nozzle Case

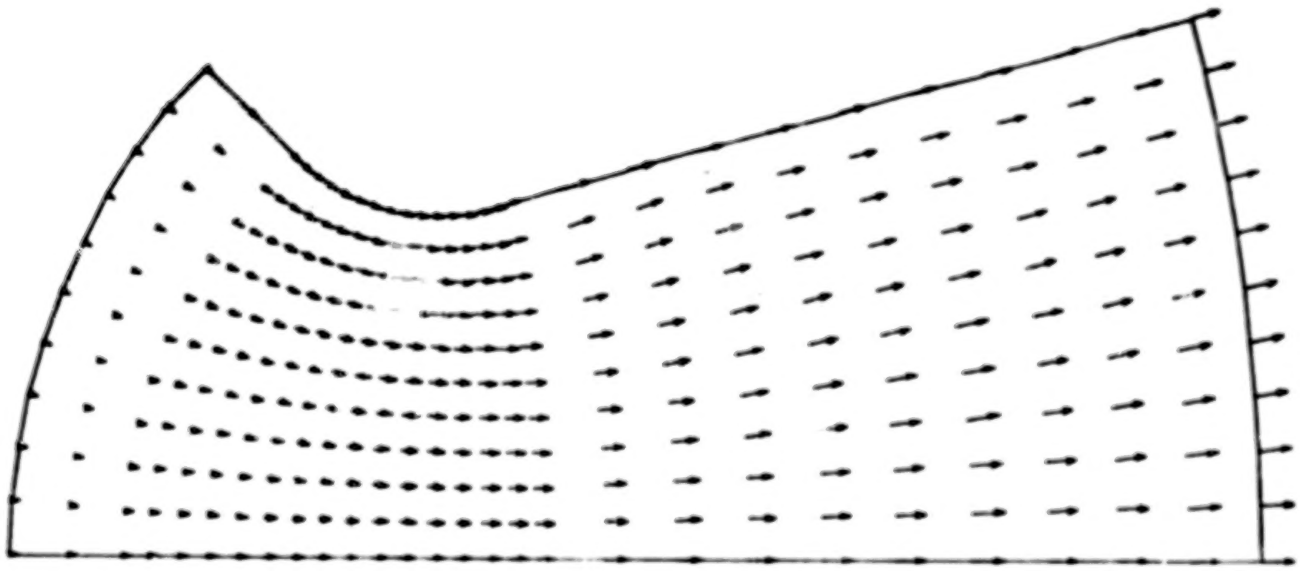


Fig. 3-2 - Velocity Vectors for Transonic Nozzle Case

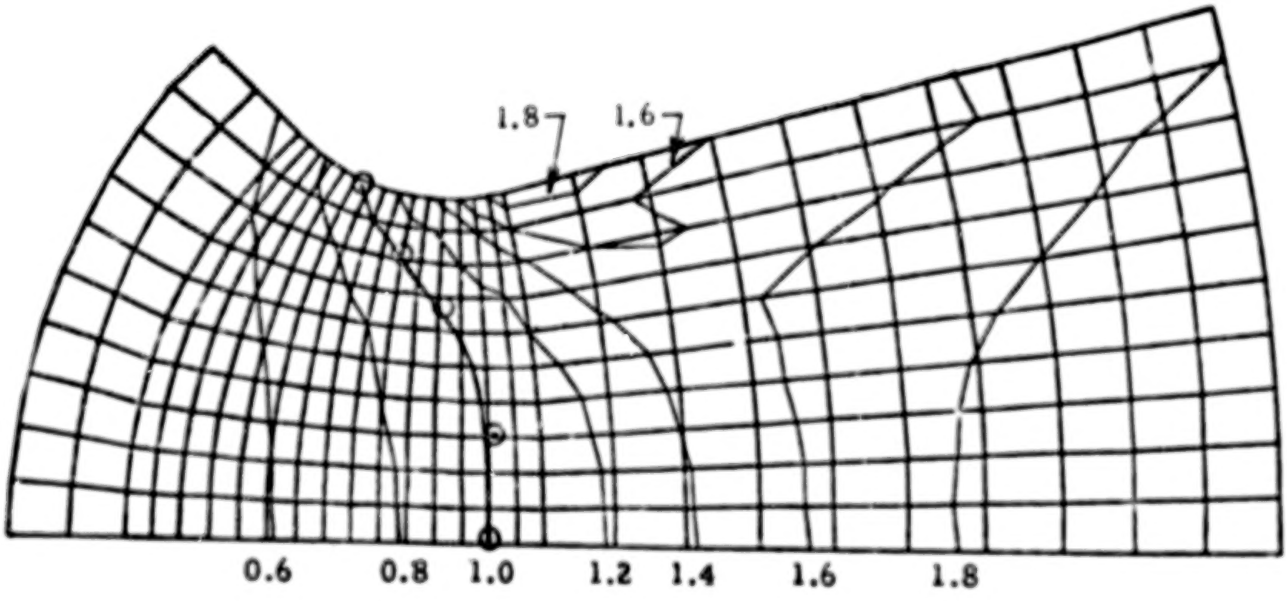


Fig. 3-3 - Mach Number Contours for Transonic Nozzle Case
(Comparison with Ref. 7 for Sonic Line)

3.2 TWO-DIMENSIONAL SHOCK CAPTURING

This case consists of the supersonic flow in a two-dimensional converging nozzle section to illustrate the "shock capturing" mode of the method. Figure 3-4 shows the mesh and geometry for the prob em. The upstream boundary condition is a specified Mach number = 4, with tangency being enforced on the centerline and the upper wall. Steady state Mach number contours are plotted in Fig. 3-5 and compared to a previous shock capturing forward marching solution. The MacCormack method was selected for use in GIM since the previous shock capturing technique (Ref. 8) used this algorithm. The figure shows the centerline Mach number versus axial distance plotted as the solid line and triangles. The GIM calculations are shown by the triangles and captures the shock well with the usual amounts of smearing that occur with this approach. The upper wall profile is plotted as dashed lines and circles. Comparison with the previous MacCormack solutions is seen to be good. The previous SCT solutions were generated via a supersonic forward marching code, while the GIM results are generated with an unsteady relaxation code.

3.3 INTERNAL BALLISTICS SIMULATION

This problem consists of an internal ballistics computation. The real propellant burning processes and grain regression were not simulated, but rather a model was used with wall mass injection at a specified rate. The sketch at the bottom of Fig. 3-6 illustrates the model. A solid wall was used as the left-most boundary, with an injection distribution from the "top" boundary and flow tangency along the centerline. Upper wall and centerline Mach number contours are plotted at steady state versus axial distance. The flow is seen to be basically one-dimensional in the straight section of the region. The wall Mach number then rapidly decreases as the conical section is approached, and then expands again up to approximately 1.35. A sharp decrease is then experienced as the flow tries to slow down through a relatively flat section of the "nozzle." Rapid expansion through the conical section is then the final regime of flow. At this time, there are no data or previous computation of

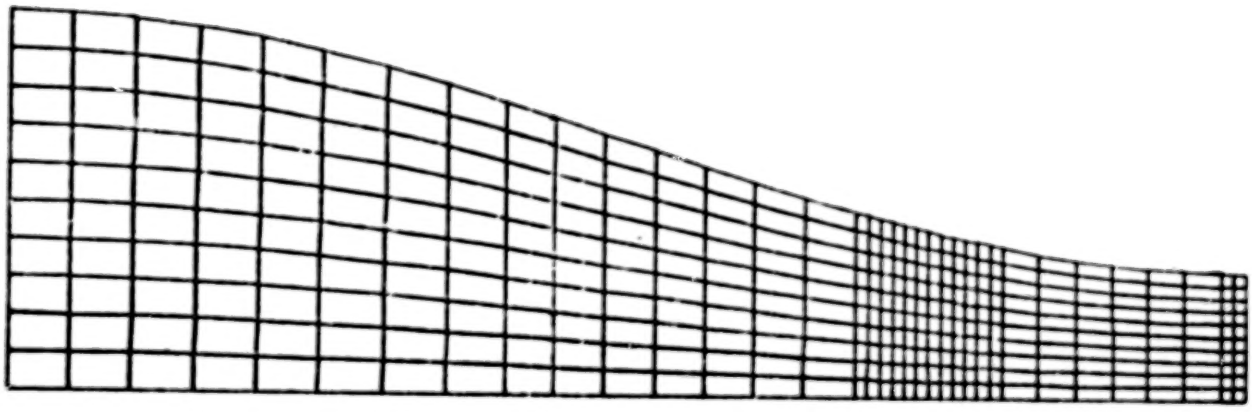


Fig. 3-4 - Geometry for Converging Nozzle Case

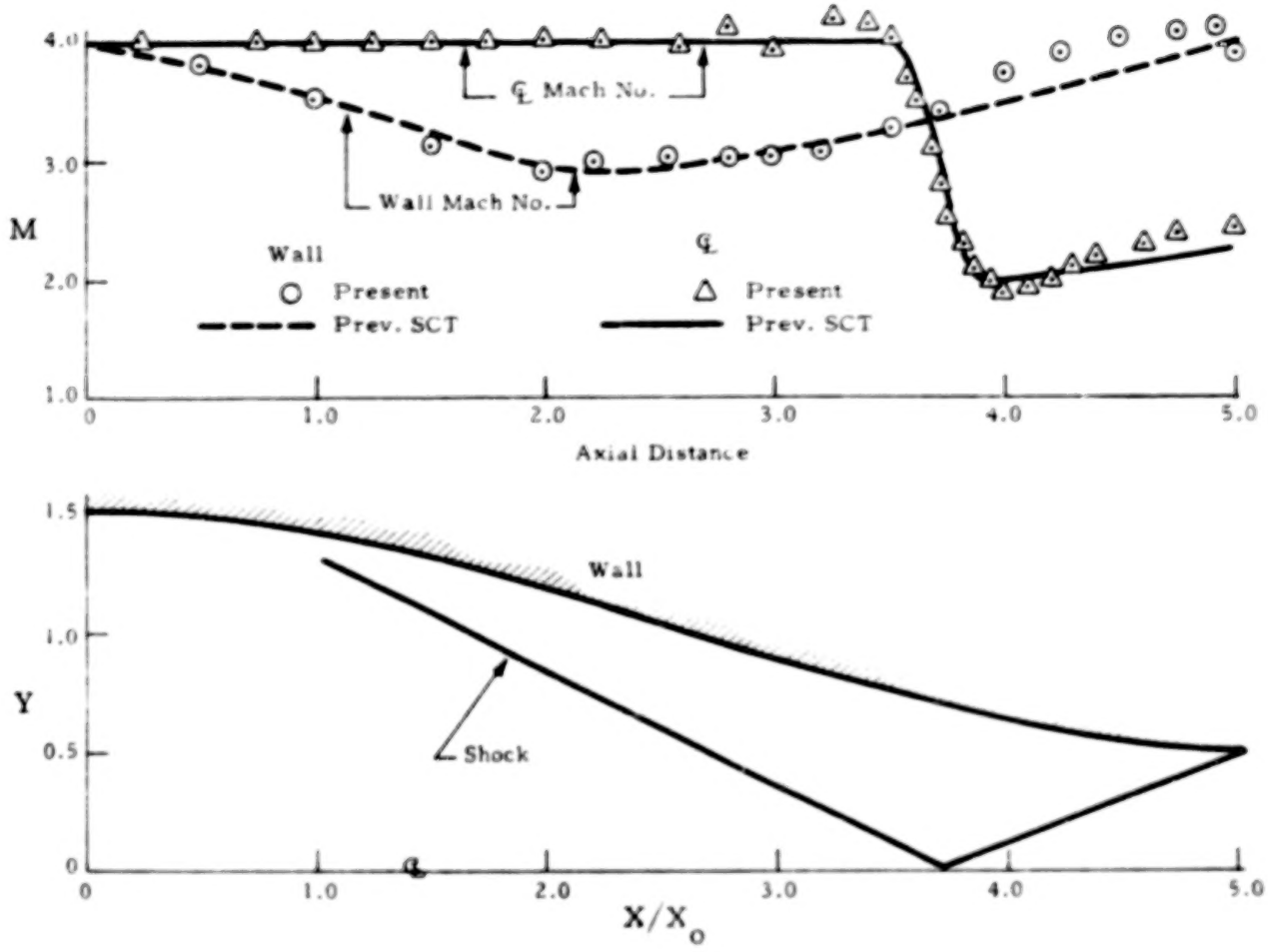


Fig. 3-5 - Mach Number Contours for Convergent Nozzle Case (X_o = .3048 m)

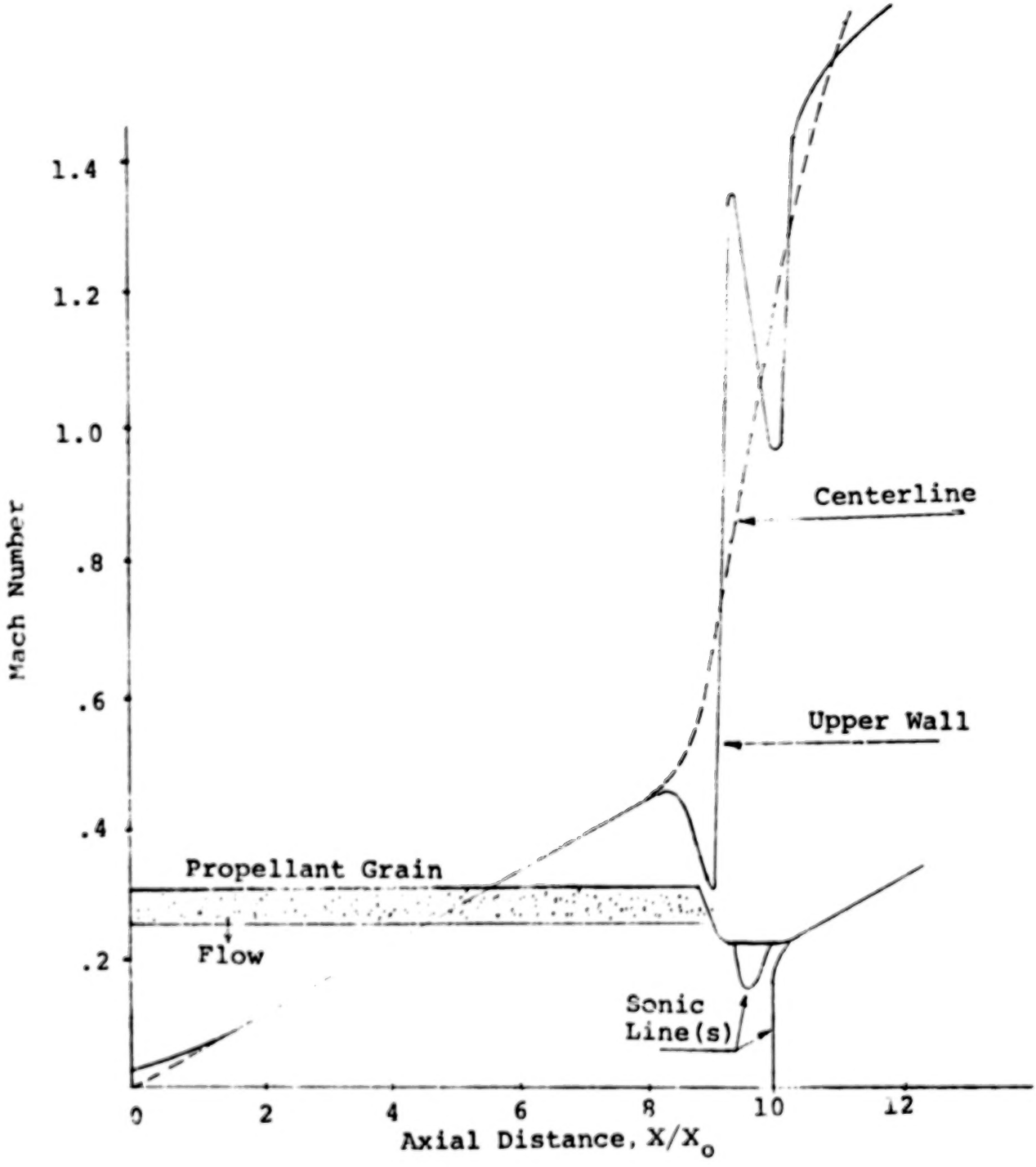


Fig. 3-6 - Two-Dimensional Internal Ballistics Computation Mach Number vs Axial Distance (X_o = .3048 m)

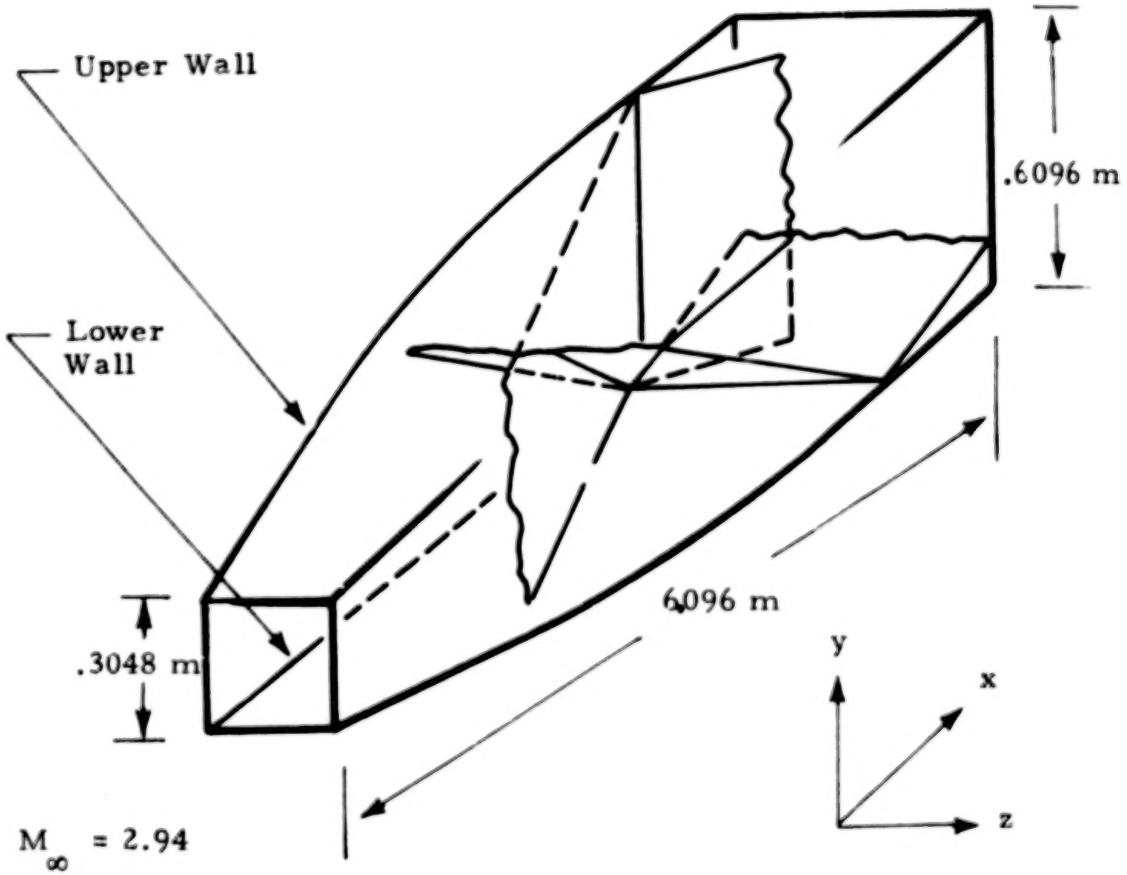
this type to be used for comparison, but the solution appears to be physically correct.

3.4 THREE-DIMENSIONAL SUPERSONIC FLOW

Next we consider three-dimensional inviscid flow in a square nozzle as shown in Fig. 3-7. The in-flow Mach number is 2.94 ($\gamma = 1.4$) at the "entrance" of the nozzle. The first 10 X_o consists of an expansion region with the final 10 X_o being a constant area section. The problem thus consists of a supersonic expansion-recompression with two intersecting shock waves as depicted in Fig. 3-7. This problem has been solved previously by personnel in the Advanced Technology Laboratory (Ref. 9) and at Lockheed-Huntsville. Both of these solutions utilize a supersonic forward marching technique with a relatively fine mesh (21 x 18 x several hundred marching steps). We have computed this problem using a two-step MacCormack algorithm with the unsteady GIM code with an extremely coarse mesh (11 x 11 x 41). This required 540 time steps to reach steady state from a "cold" start, i.e., no information about the flow structure.

The very coarse grid was used for this case to: (1) check out and demonstrate the basic three-dimensional GIM code without an excessive amount of computer time; and (2) to test the stability of the scheme for a very coarse grid. The GIM solution is shown in Fig. 3-8 and compared with the ATL solution. Shown are plots of pressure versus axial distance at the upper and lower walls in the symmetry plane of the nozzle (see Fig. 3-7). The calculations show excellent agreement in the expansion region but deviate considerably in the shockwave region. They do, however, exhibit the proper behavior with the quantitative disagreement due possibly to the very coarse grid in the GIM solution or a computation problem with the ATL solution.

Figure 3-9 is a contour map of pressure in the x-y plane at the outer wall ($z = 2.0$) of the nozzle. The plot shows the shock wave and reflection clearly even though it is smeared considerably by the crude grid. This case is shown to demonstrate the correctness and applicability of the GIM approach to complex fluid dynamic computations. Refinement of the solution is now a matter of grid size and computer time.



$(z, y) = 1.5 - 0.5 \sin \left(\frac{\pi}{2} + \frac{\pi x}{10}\right), x \leq 10$

$(z, y) = 2 , x > 10$

(All units dimensionless)

Fig. 3-7 - Square Nozzle Three-Dimensional Case Expansion-Recompression

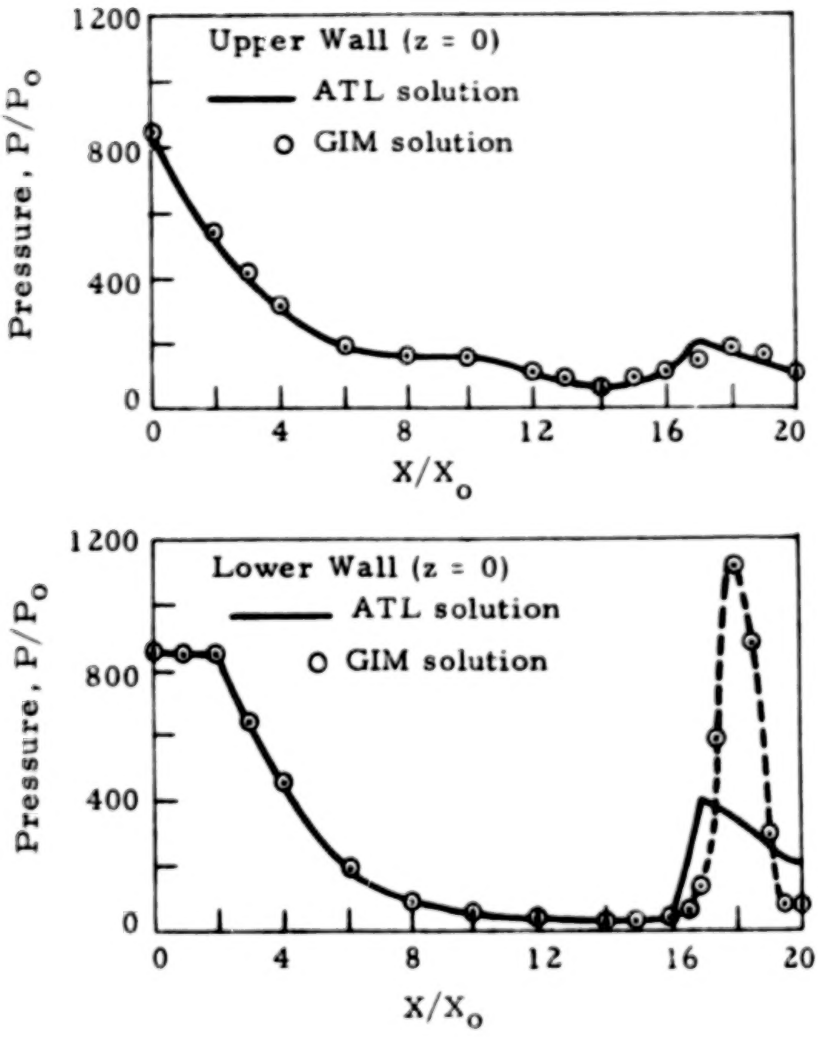


Fig. 3-8 - Pressure vs Axial Distance at Upper and Lower Wall of Square Nozzle – Comparison with ATL Solution (Ref. 9)
(X_o = .3048 m, P_o = 47.87 N/m^2)

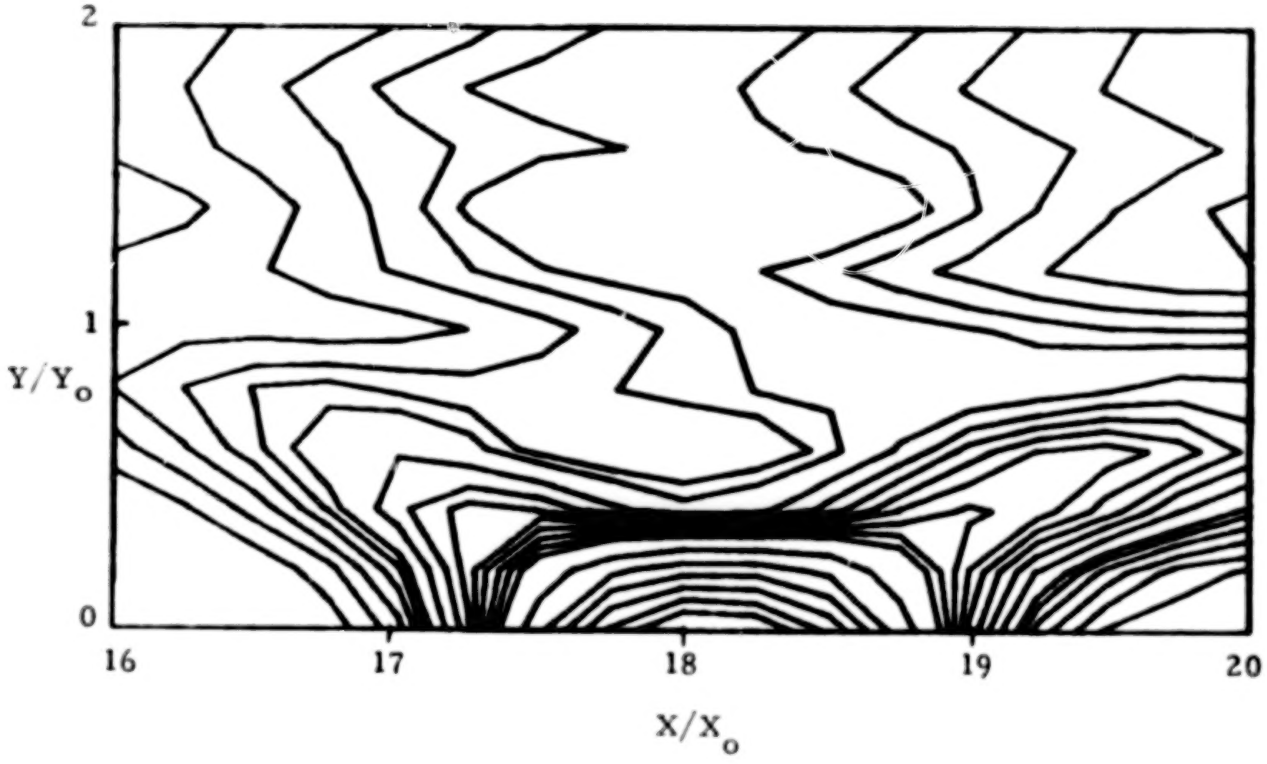


Fig. 3-9 - Steady Pressure Contours in x-y Plane Showing Shock Reflection (Coarse grid 11 x 11 x 41) GIM Code Solution ($X_o = Y_o = .3048$ m)

3.5 THREE-DIMENSIONAL TRANSONIC FLOW

This problem involves the computation of three-dimensional flow in a spinning plug nozzle (SPN). The spinning plug nozzle consists of a tubular section which necks down to a smaller tubular section, as shown in Fig. 3-10. A plug is inserted into the neck-down region to form a three-dimensional convergent-divergent nozzle. The plug is symmetric about its own centerline but may be deflected from the centerline of the tubular portion, resulting in a nonsymmetric flow field. The flow includes subsonic, transonic, and supersonic regimes.

The flow field was computed with the GIM code using 11,628 nodes. The results of the computation are presented in Figs. 3-11 and 3-12. These results were used in a contractual study conducted by Lockheed for the U.S. Army Missile Research & Development Command (MIRADCOM).

Figure 3-11 shows the Mach number contours at steady state at the upper and lower walls of the configuration as shown in Fig. 3-10. Figure 3-12 is the corresponding pressure contours. At present there are no data to compare with for this configuration, but the results appear to be physically real. Comparison will be made when the data become available.

3.6 TWO-DIMENSIONAL SHEAR FLOWS

The flow field analyzed involved mixing the exhaust from a two-dimensional Scramjet afterbody nozzle with freestream. The problem configuration and the flow properties of the two flow streams are shown in Fig. 3-13. Both flow streams have the same value for the ratio of specific heats ($\gamma = 1.27$). This problem was solved under contract from NASA-Langley Research Center.

The grid used in GIM computation is shown in Fig. 3-14. Quadrilateral elements were used with a two-step MacCormack-type of analog. Solutions were obtained at steady state by relaxation of the unsteady equations. Figures 3-15 and 3-16 are contour maps of pressure and Mach number, respectively,

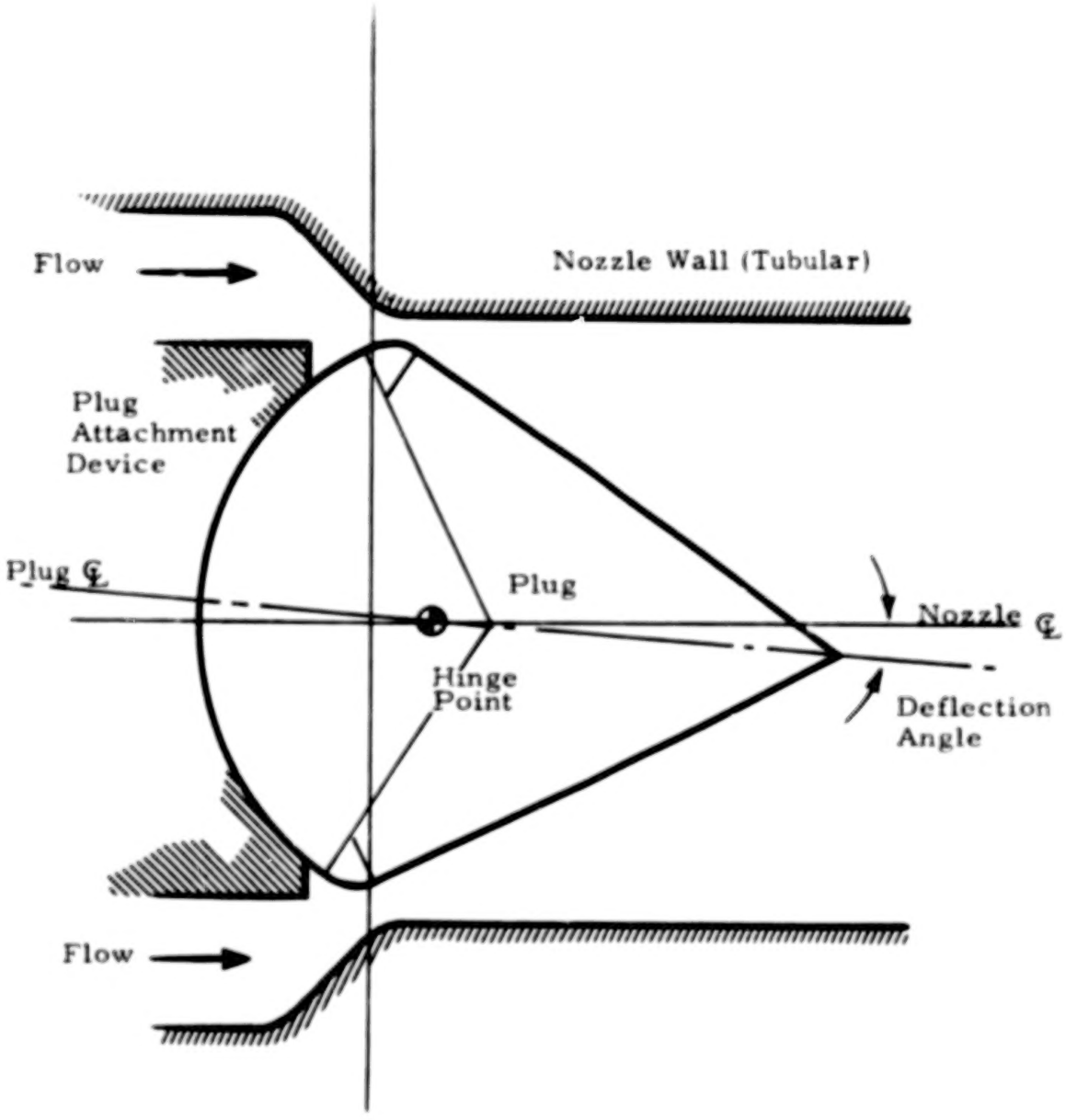


Fig. 3-10 - Cross Section of Spinning Plug Nozzle

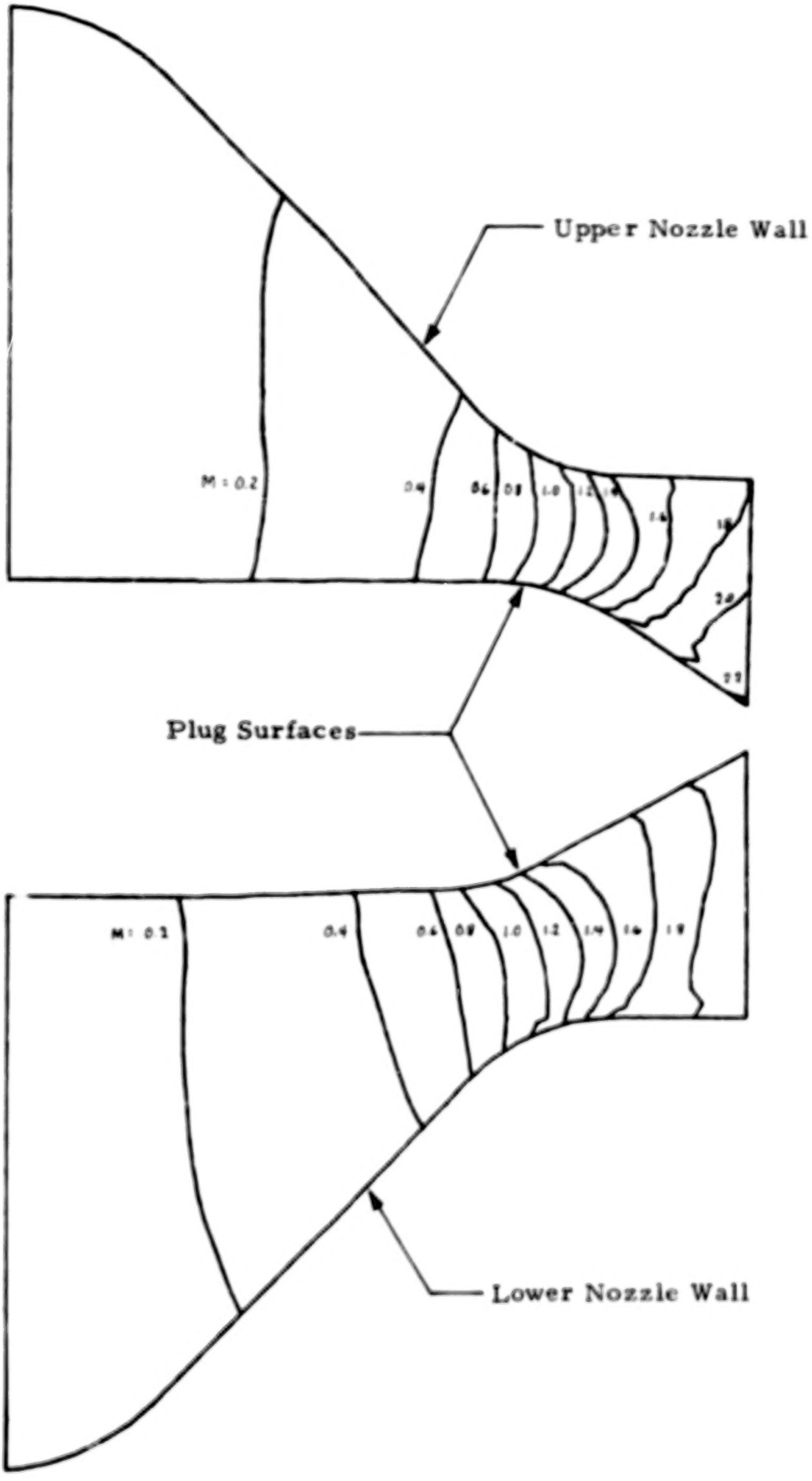


Fig. 3-11 - Mach Number Contours for Spinning Plug Nozzle

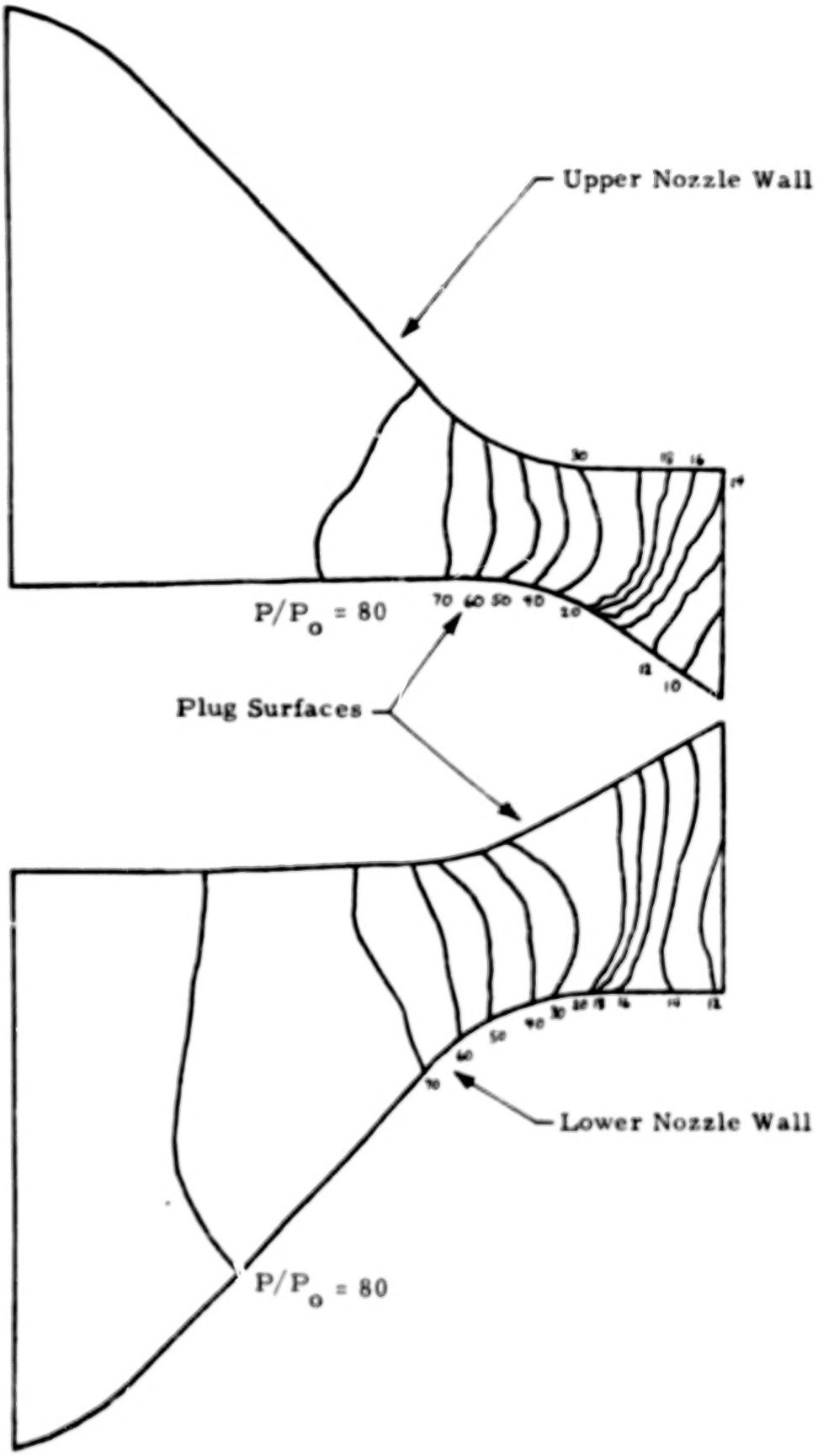


Fig. 3-12 - Pressure Contours for Spinning Plug Nozzle
(P_o = 47.87 E + 3 N/m^2)

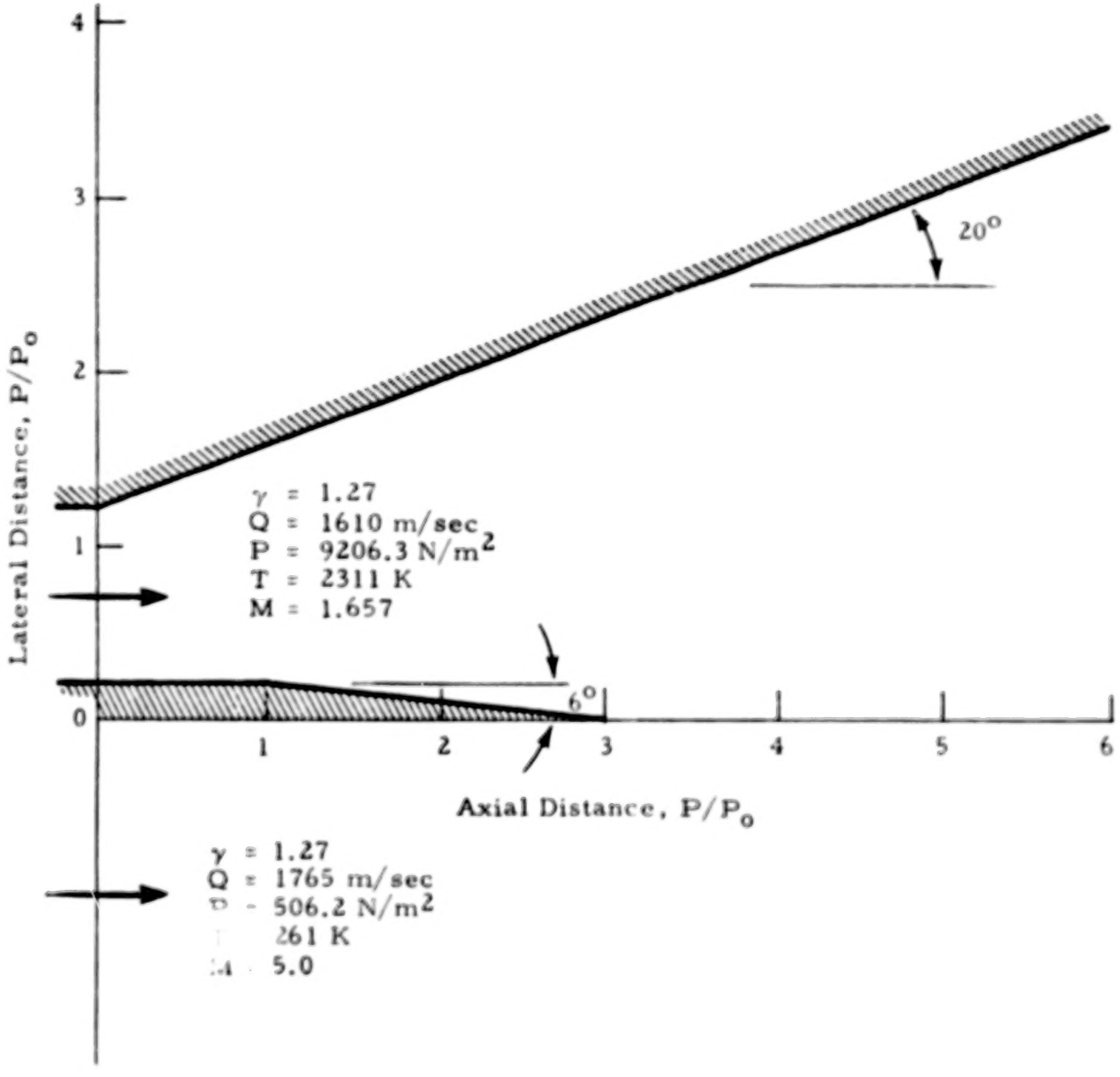


Fig. 3-13 - Two-Dimensional Shear Flow Configuration

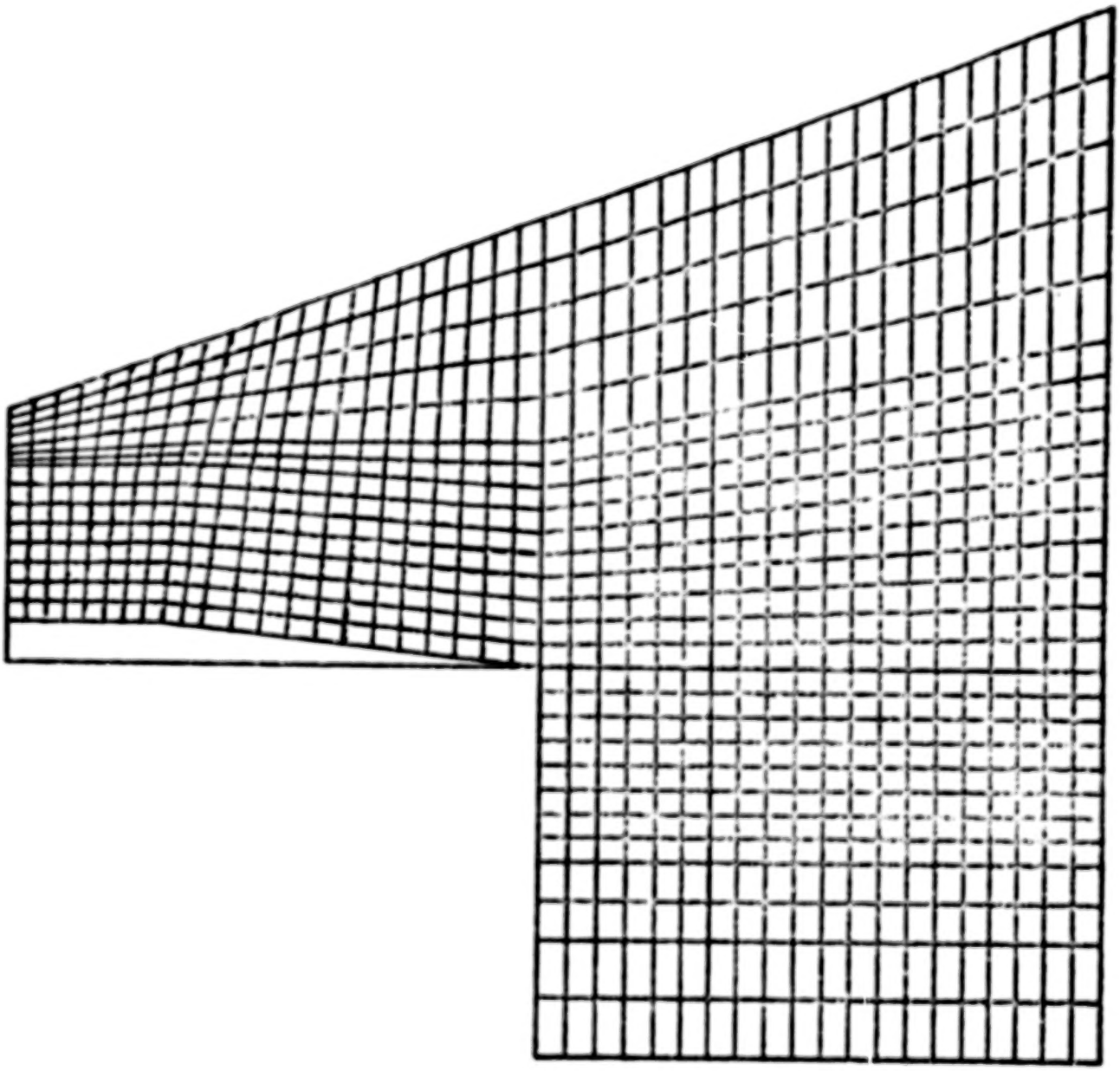


Fig. 3-14 - Finite Element Grid for Two-Dimensional Configuration

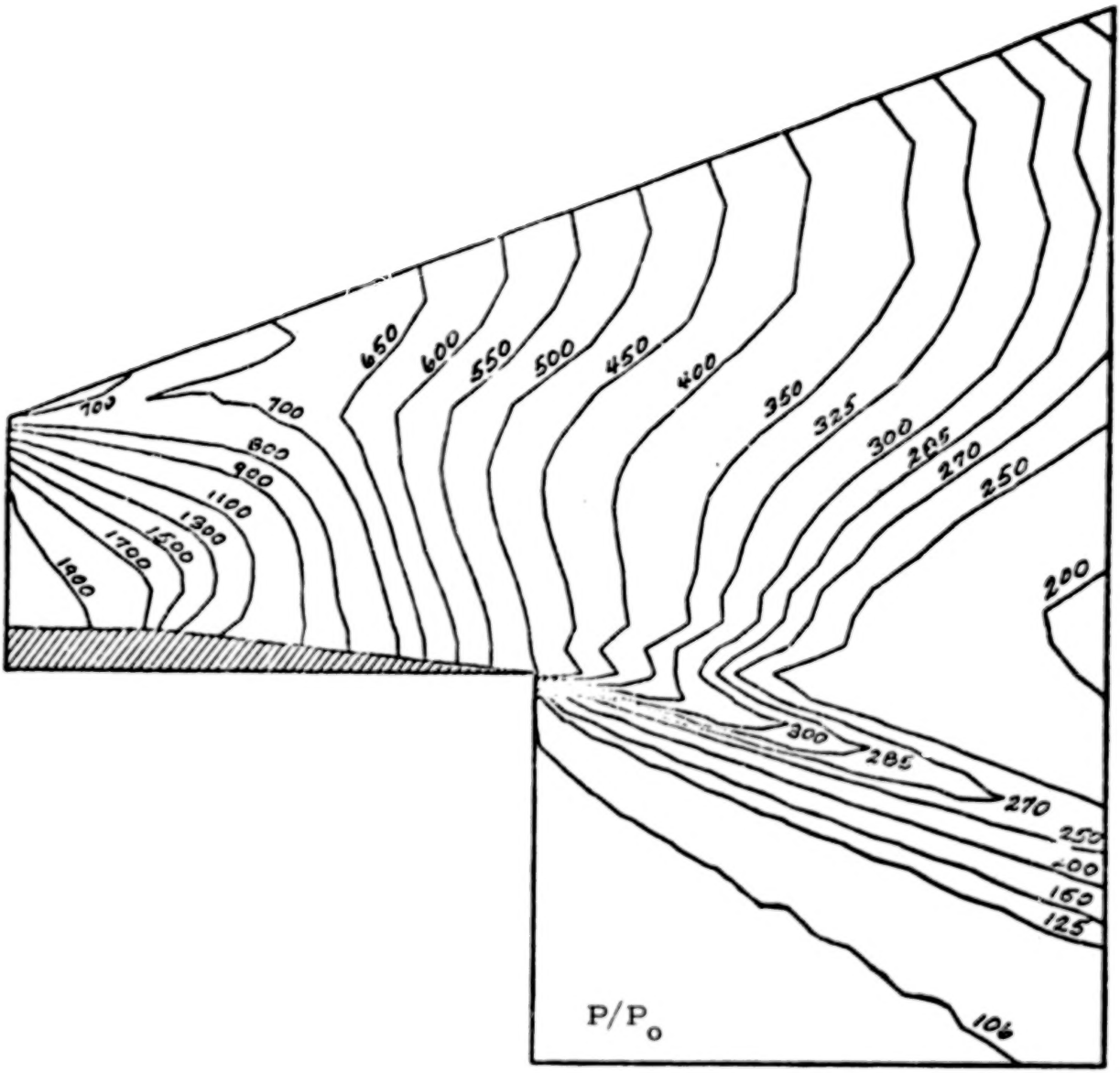


Fig. 3-15 - Pressure Contours for Two-Dimensional Shear Flow Case (P_o = 47.87 N/m^2)

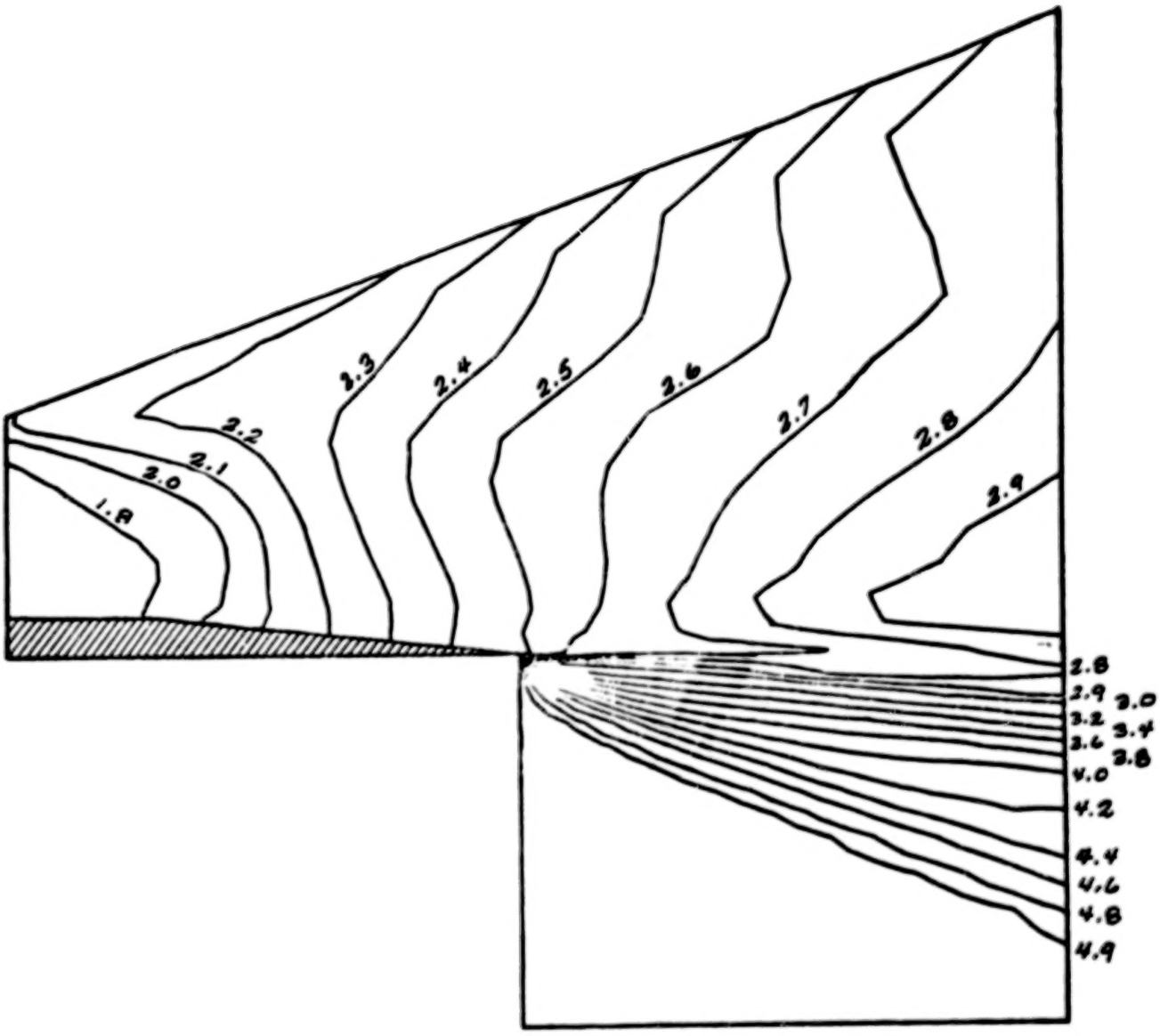


Fig. 3-16 - Mach Number Contours for Two-Dimensional Shear Flow Case

at this steady state. Figures 3-17 through 3-19 are plots of the GIM solution compared to the Seagull code solution (Ref. 10). Figure 3-17 shows vertical pressure distributions through the shear layer at three axial stations downstream of the initial mixing zone. Good agreement is seen except at the upper wall for the "close-in" stations. Figure 3-18 is the corresponding Mach number plots.

The upper wall pressure distribution is plotted in Fig. 3-19 and compared to the Seagull code and the Lockheed Method of Characteristics (MOC) code. Agreement is excellent except near $X = 3\ X_o$ where the GIM results are apparently low. This is due to the GIM code treatment of sharp corners. Seagull and MOC use a Prandtl-Meyer type expansion on all sharp corners while the GIM code solves only the differential equations.

Aside from these explainable discrepancies, the GIM code results compare favorably with the Seagull inviscid solution. The shock, although smeared by the GIM code, is located in approximately the same place in both analyses and the effects of the shear layer are apparent in the GIM code results.

3.7 THREE-DIMENSIONAL SHEAR FLOWS

A three-dimensional shear layer, resulting from the interaction of a nozzle exhaust stream with the free stream, both beneath and beside the nozzle, was computed using the GIM code. The configuration, shown in Fig. 3-20 consists of a rectangular nozzle suspended below a body or wing.

Referring to Fig. 3-20, the nozzle has a sharp 20 deg upward turn at $X = 0$, a sharp 6 deg turn downward at $X = X_o$, and a sharp 6 deg turn outward along the dashed line located on the sidewall. The rectangular nozzle exit plane is inclined 30 deg from vertical. The body has a sharp 20 deg turn upward at $X = X_o$.

The problem was analyzed in three parts: (1) the nozzle; (2) the external flow upstream of the nozzle exit, and (3) the entire flow downstream of the exit

plane. The nozzle flow field, which is three-dimensional, was computed using the GIM code. It was computed using 2160 nodes and required 570 iterations to relax to the steady solution. The external flow was computed using a Prandtl-Meyer expansion at the 20 deg sharp turn. The nozzle exit plane conditions and the freestream expansion were used as input boundary conditions for the downstream shear layer region.

The shear layer region was analyzed using 7094 nodes and converged in 260 iterations. A single grid surface is shown in Fig. 3-21 to illustrate the grid density and alignment used in the analysis. The results of the analysis are presented in Figs. 3-22 through 3-27. A cross section of the problem is inserted in each figure to indicate which areas of the flow field the data represent.

Figure 3-22 presents vertical pressure profiles in the plane of symmetry for three downstream stations. The profiles illustrate the downward movement and dissipation of the shear layer. Figure 3-23 presents pressure profiles along the upper wall at constant values of X. The gradients are more severe due to the expansion of the free stream, and the 20 deg turn, from 105.8 P_o to 7.5 P_o. The shear layer along the upper wall is shown to move outward and dissipate.

Figures 3-24 through 3-26 present pressure contours at three axial stations. An outline of the nozzle has been superimposed for reference. Figure 3-24 shows the thin shear layer just downstream of the exit plane and the freestream pressure gradient due to the Prandtl-Meyer expansion. Figures 3-25 and 3-26 illustrate how the external flow rolls upward and outward when not constrained by the nozzle walls. The dissipation and outward movement of the shear layer are also illustrated. Figure 3-27 presents the pressure contours on the upper wall, both inside and outside the nozzle. The nozzle sidewall with its 6 deg outward turn is apparent in this figure. The freestream Prandtl-Meyer expansion appears as a step function. There are no data to compare this complex three-dimensional solution, but again from physical insight, the results appear at least qualitatively correct.

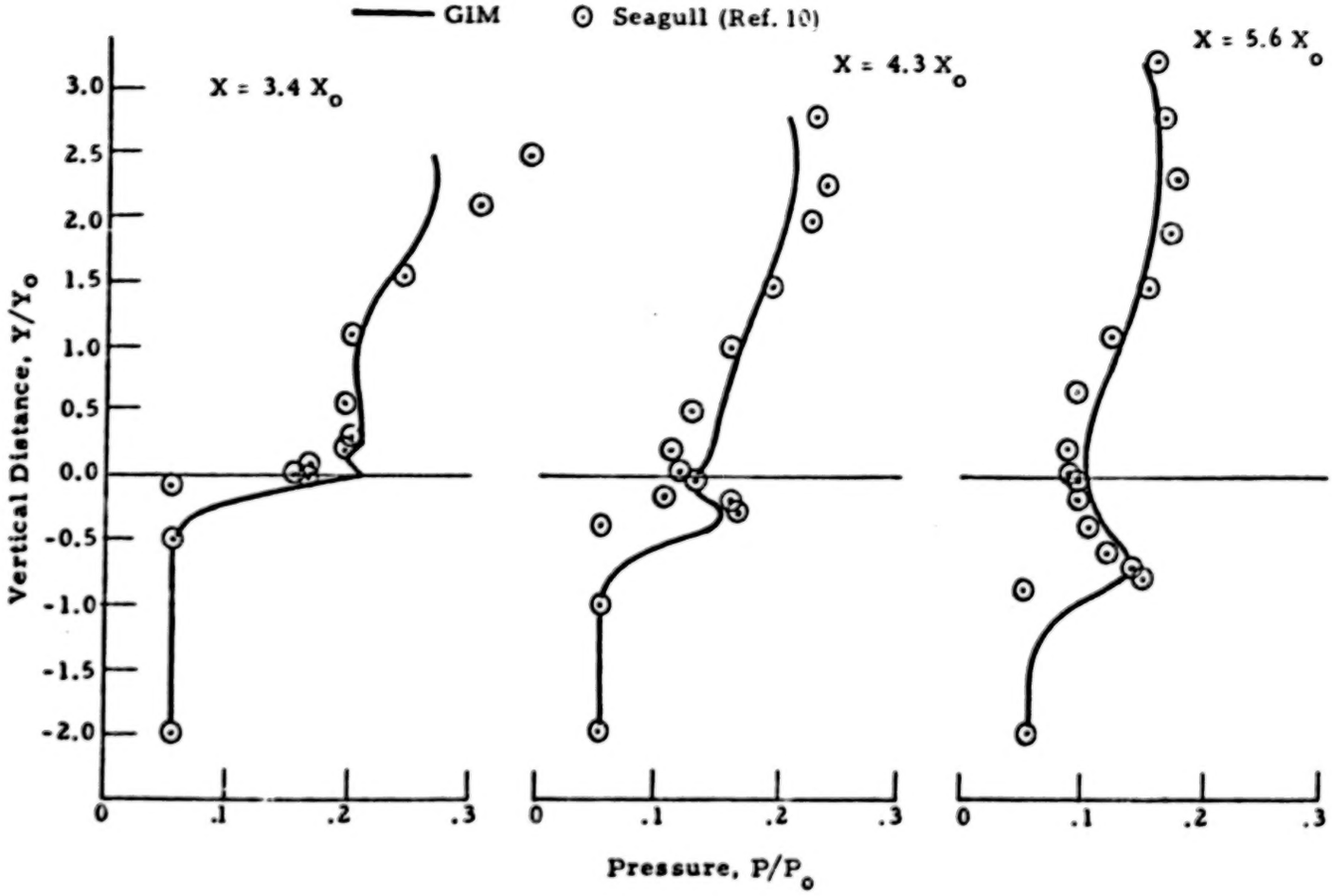


Fig. 3-17 - Vertical Pressure Distributions at Three Axial Stations ($X_o = Y_o = .3048$ m, $P_o = 9206.3$ N/m^2)

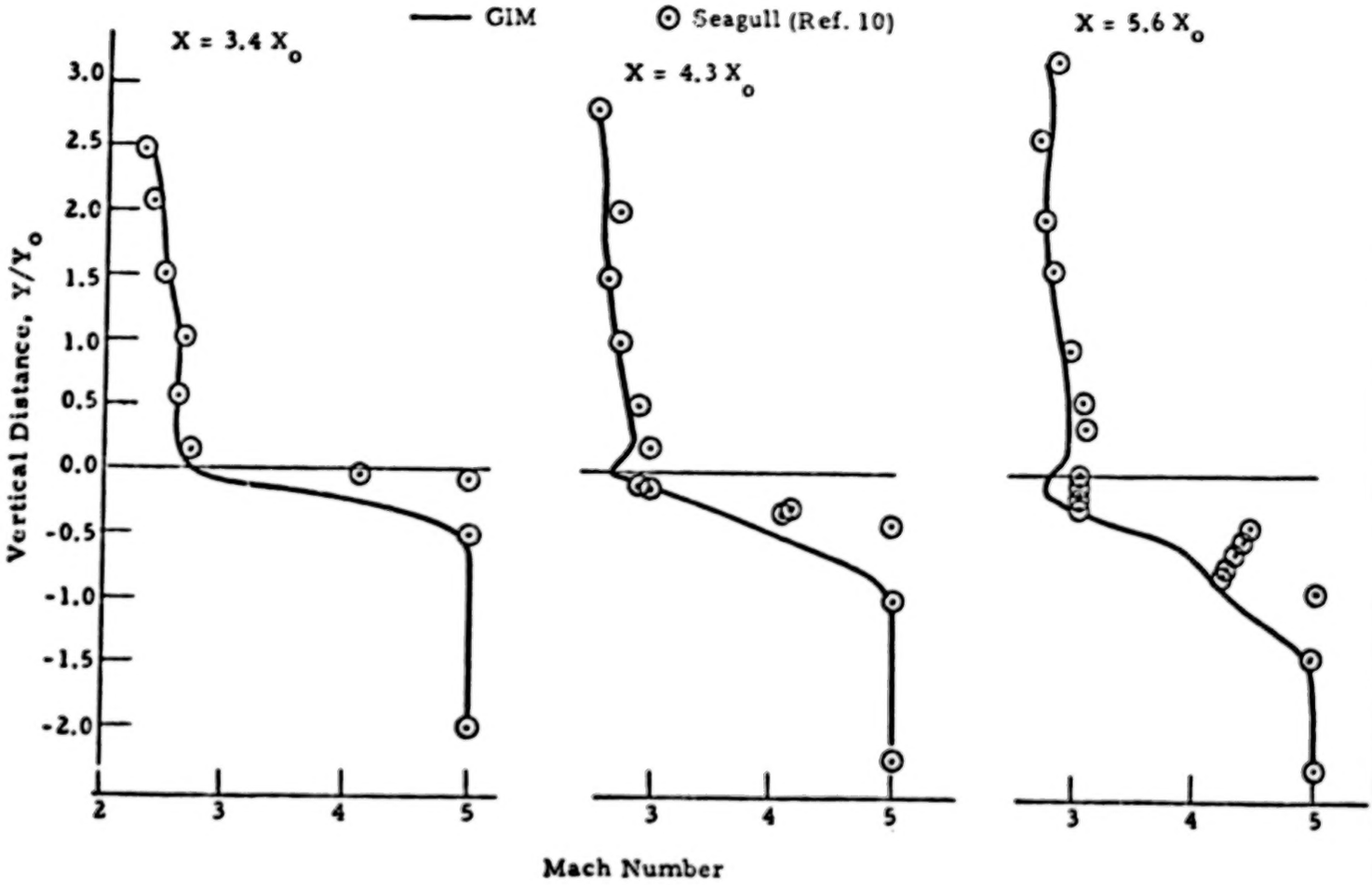


Fig. 3-18 - Mach Number vs Vertical Distance ($X_o = Y_o = .3048$ m)

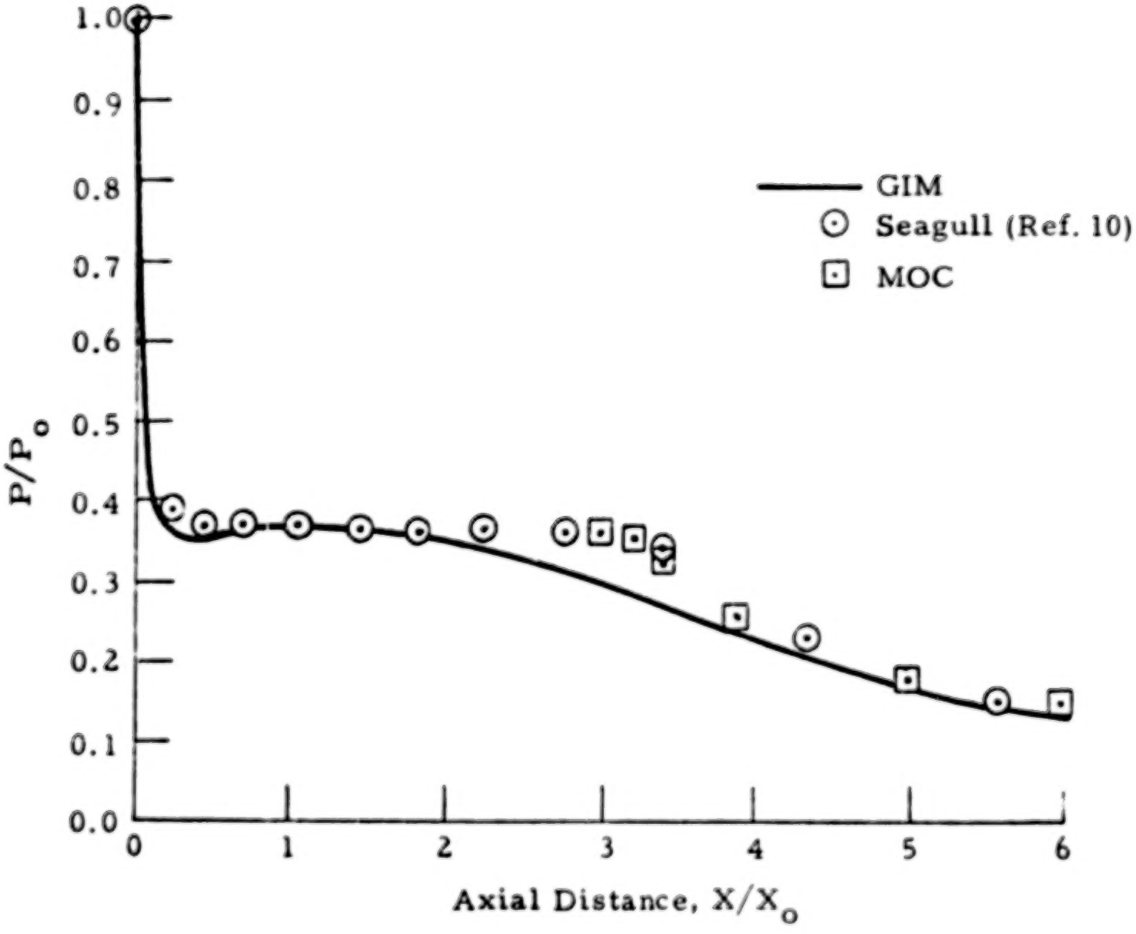


Fig. 3-19 - Upper Wall Pressure vs Axial Distance
(X_o = .30[illegible] m)

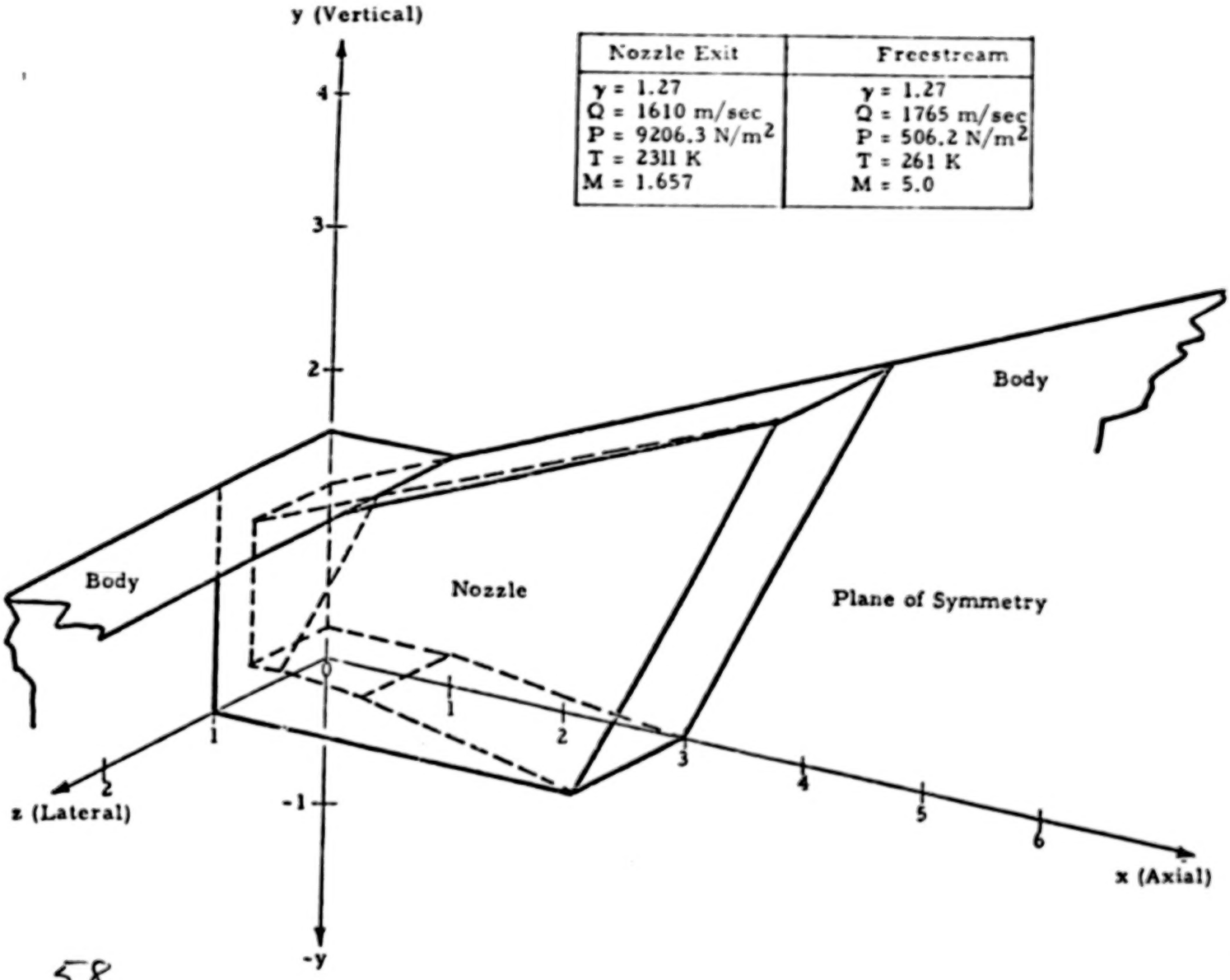


Fig. 3-20 - Three-Dimensional Outboard Module Configuration

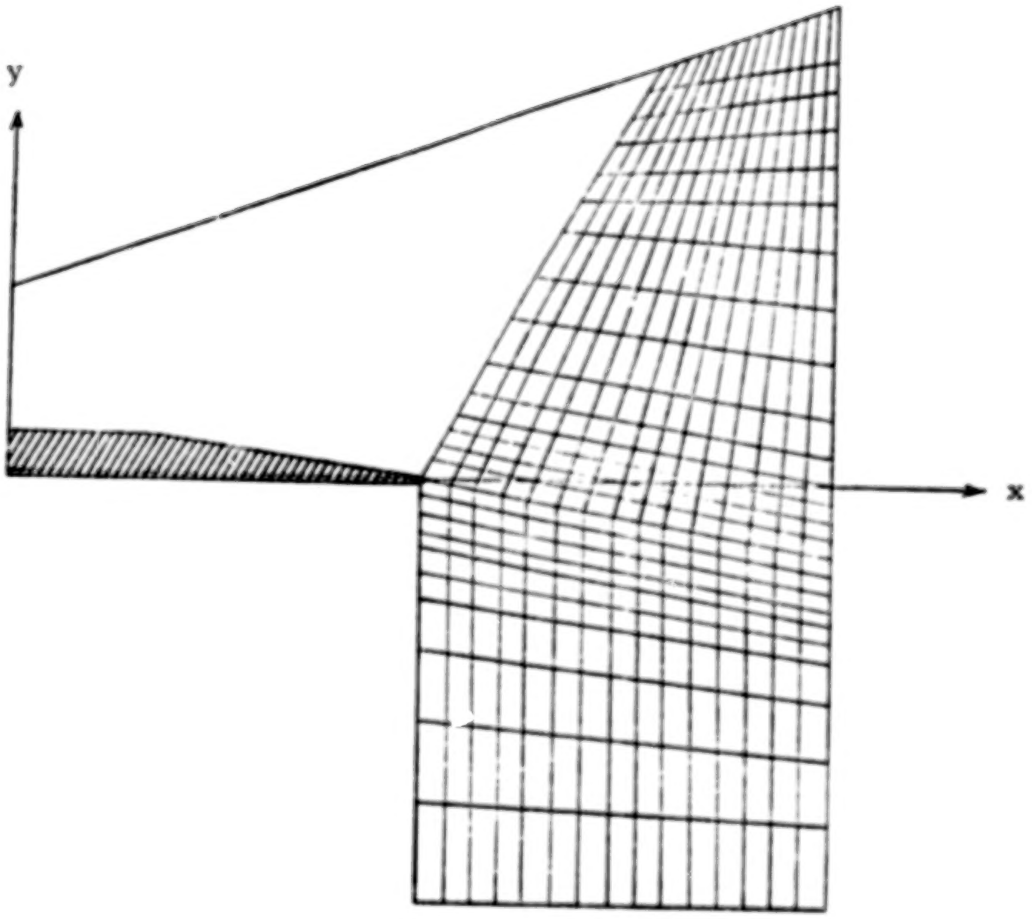


Fig. 3-21 - Nodal Distribution for Three-Dimensional Shear Layer
(z = 0 Plane Only)

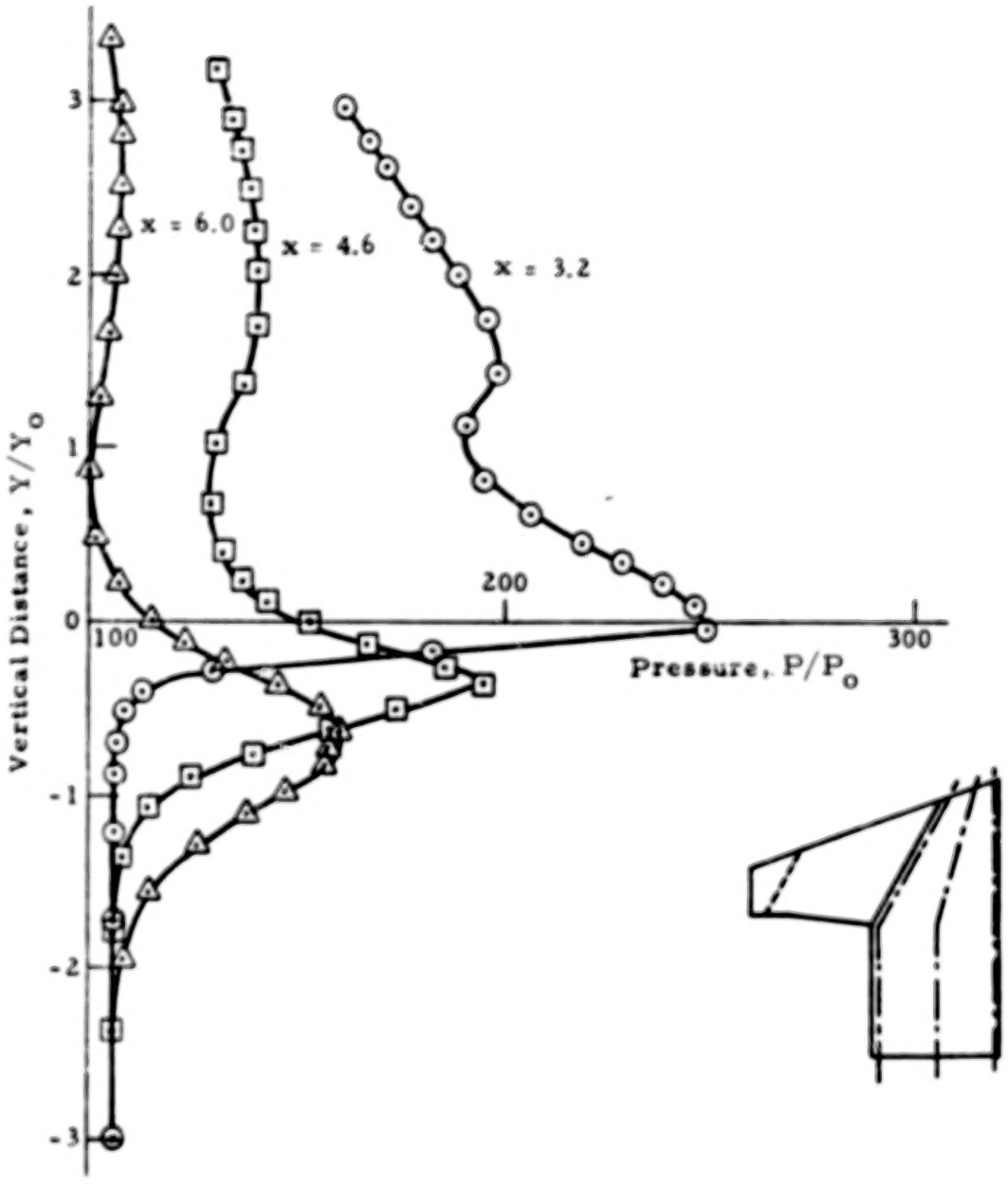


Fig. 3-22 - Vertical Pressure Distribution at Several Axial Stations (Y_o = .3048 m, P_o = 47.87 N/m^2)

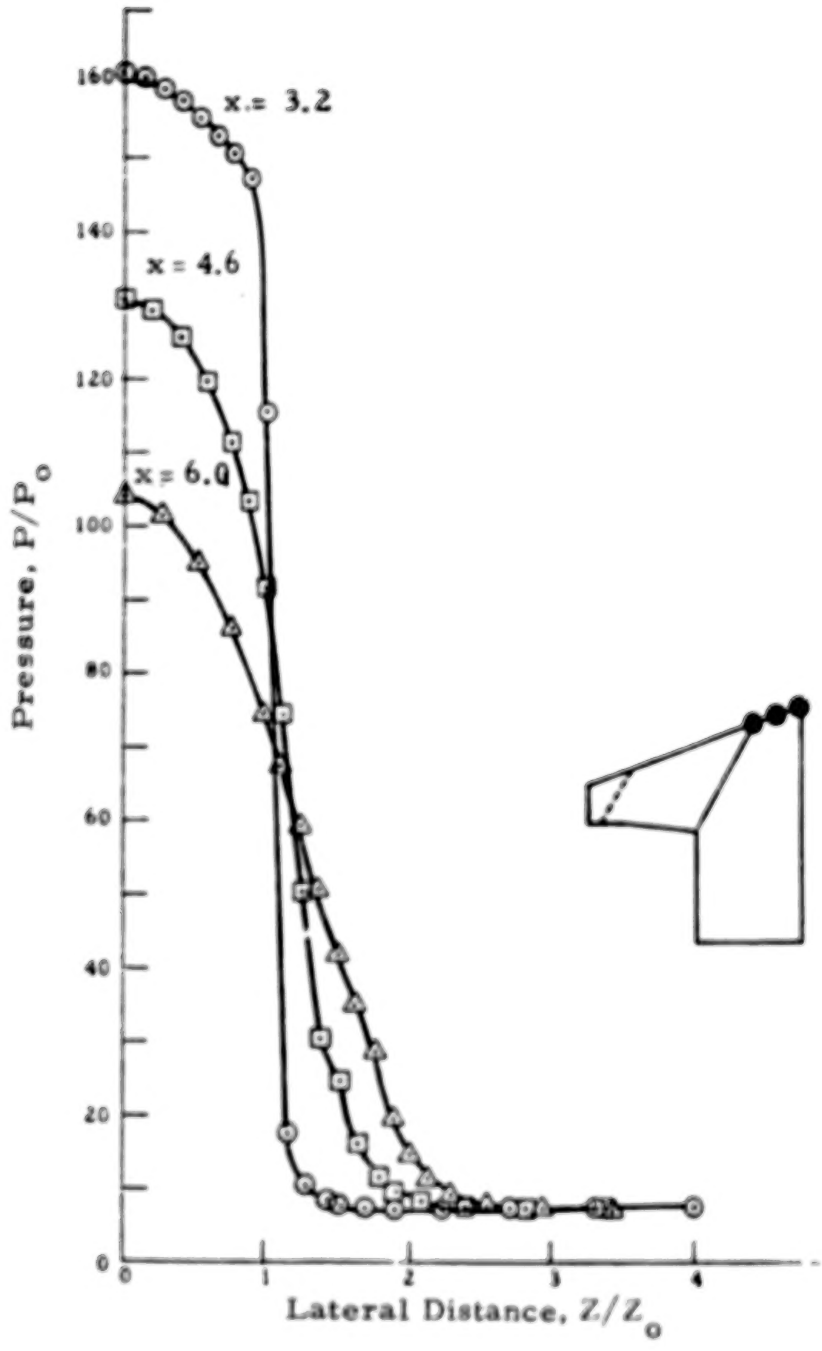


Fig. 3-23 - Lateral Pressure Distribution at Several Axial Stations (Z_o = .3048 m, P_o = 47.87 N/m^2)

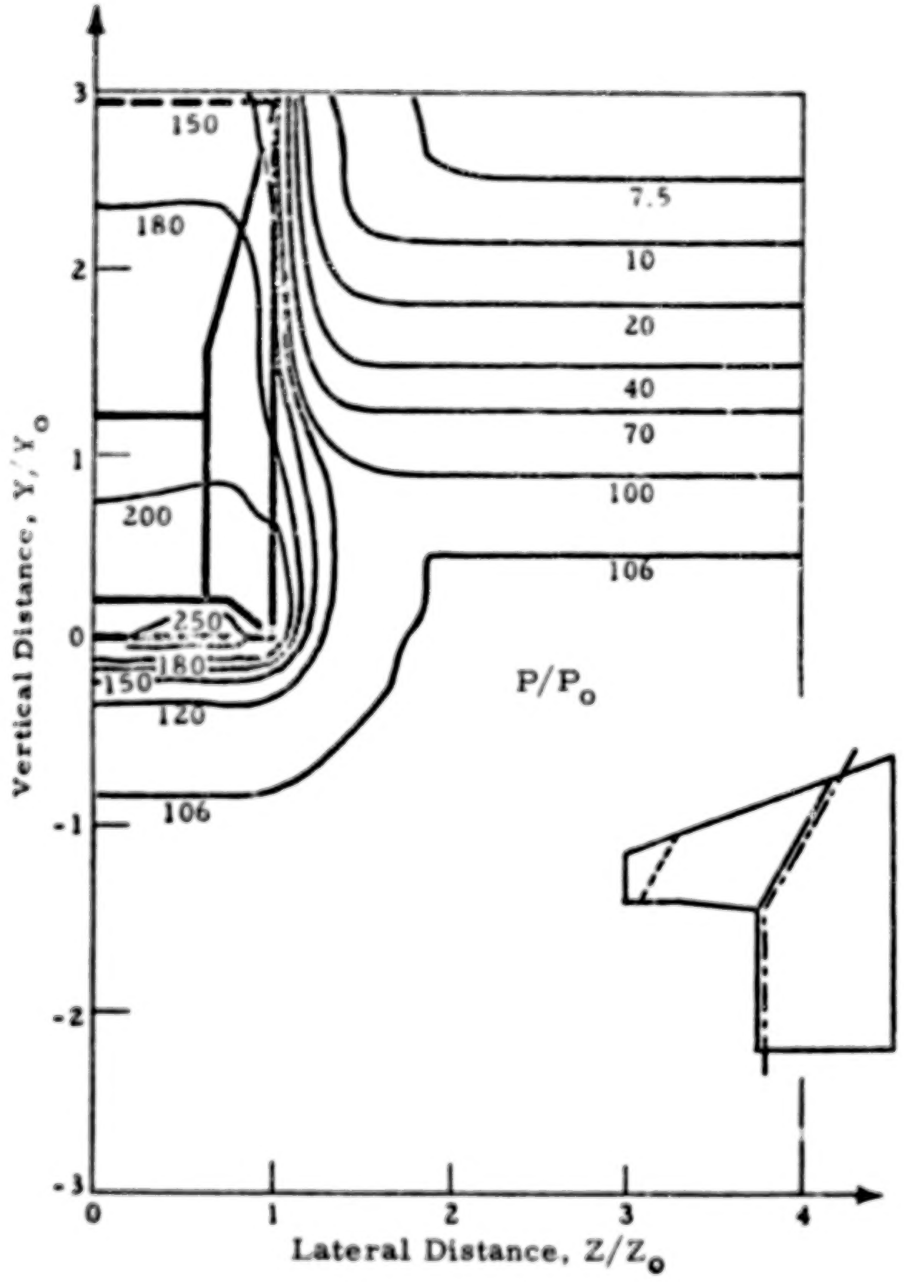


Fig. 3-24 - Pressure Contours at X = 3.2 X_o
(X_o = Y_o = Z_o = .3048 m, P_o = 47.87 N/m^2)

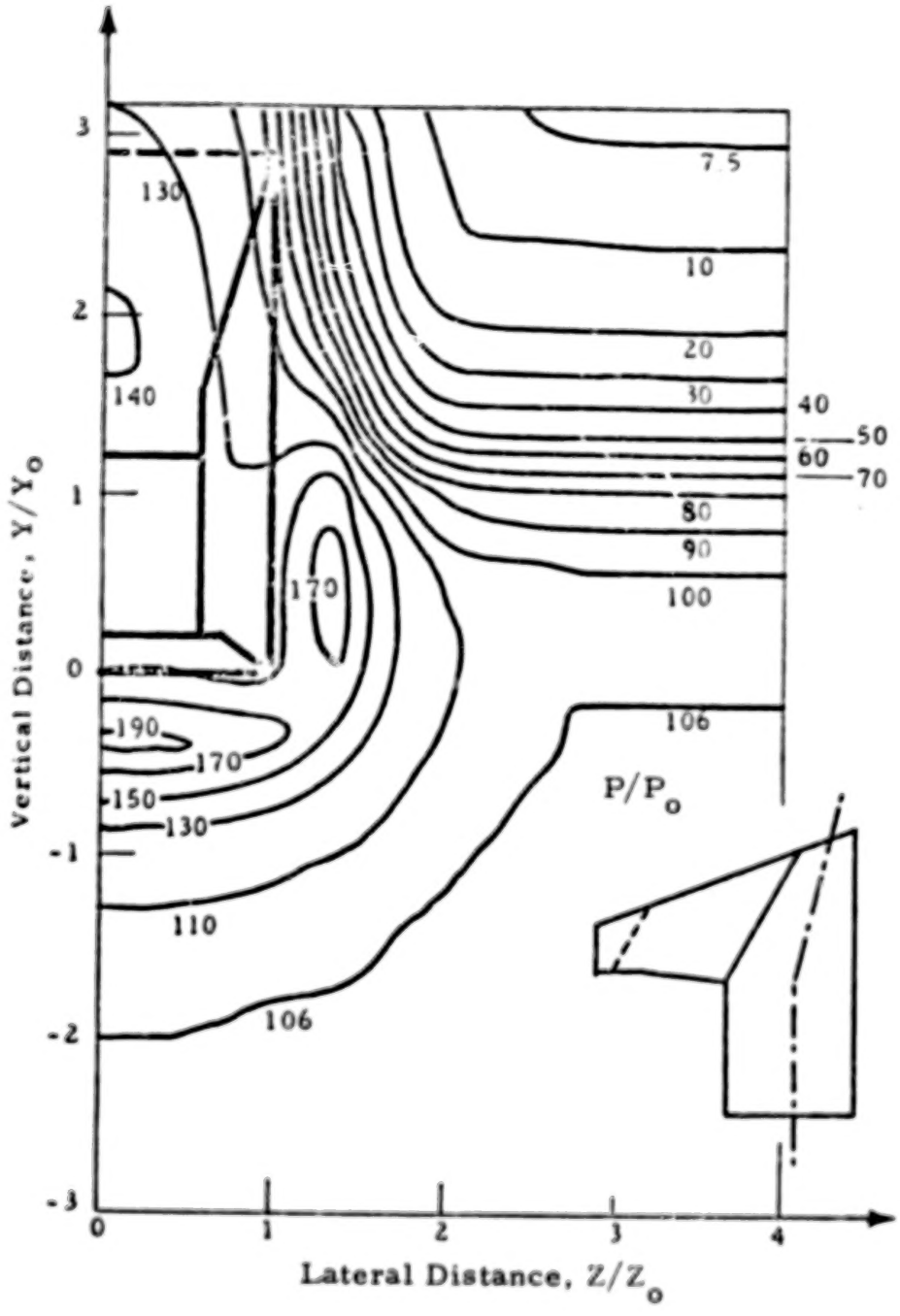


Fig. 3-25 - Pressure Contours at $X = 4.6\ X_o$
($X_o = Y_o = Z_o = .3048$ m, $P_o = 4787\ N/m^2$)

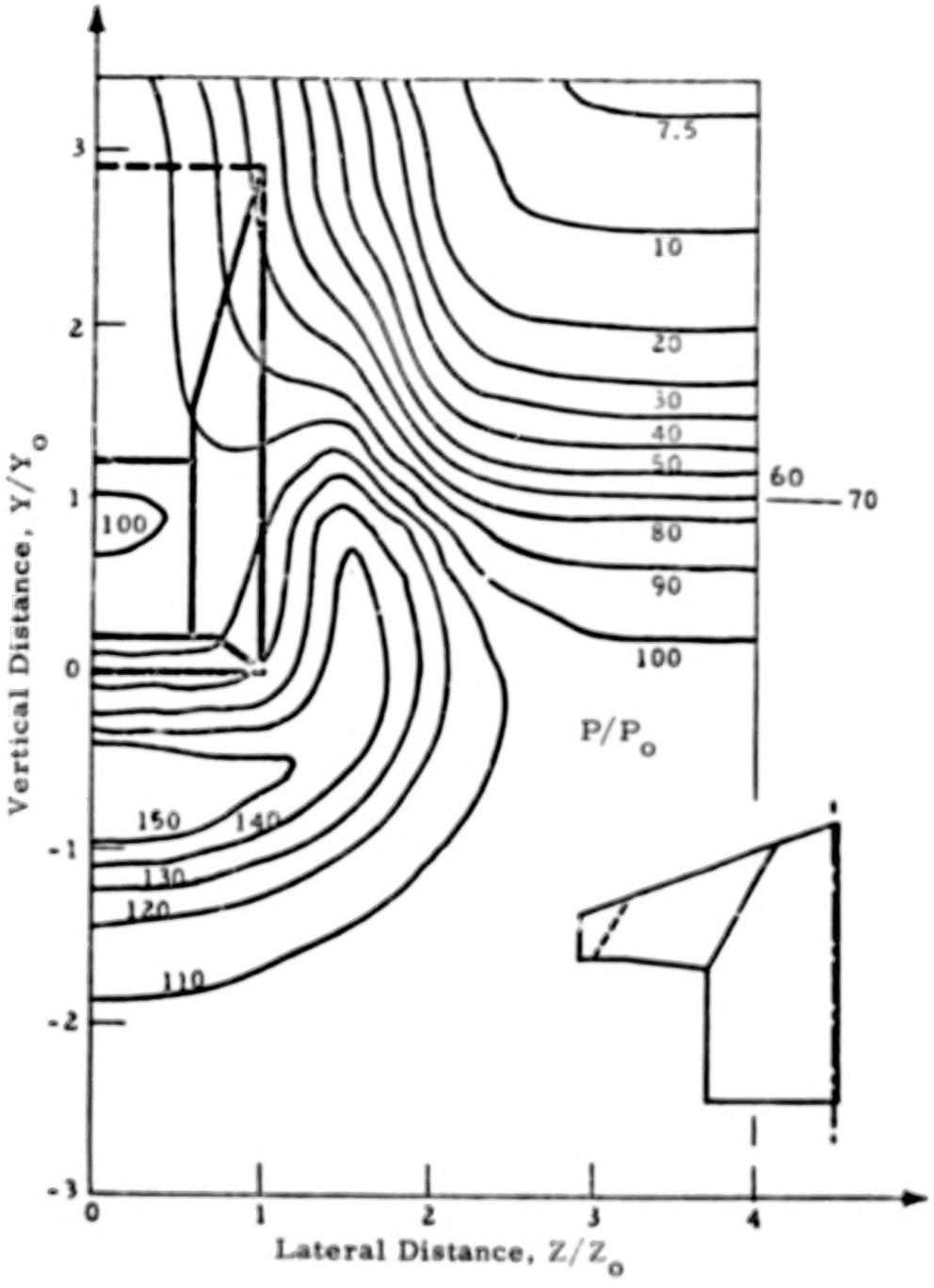


Fig. 3-26 - Pressure Contours at X = 6.0 X_o
(X_o = Y_o = Z_o = .3048 m, P_o = 47.87 N/m^2)

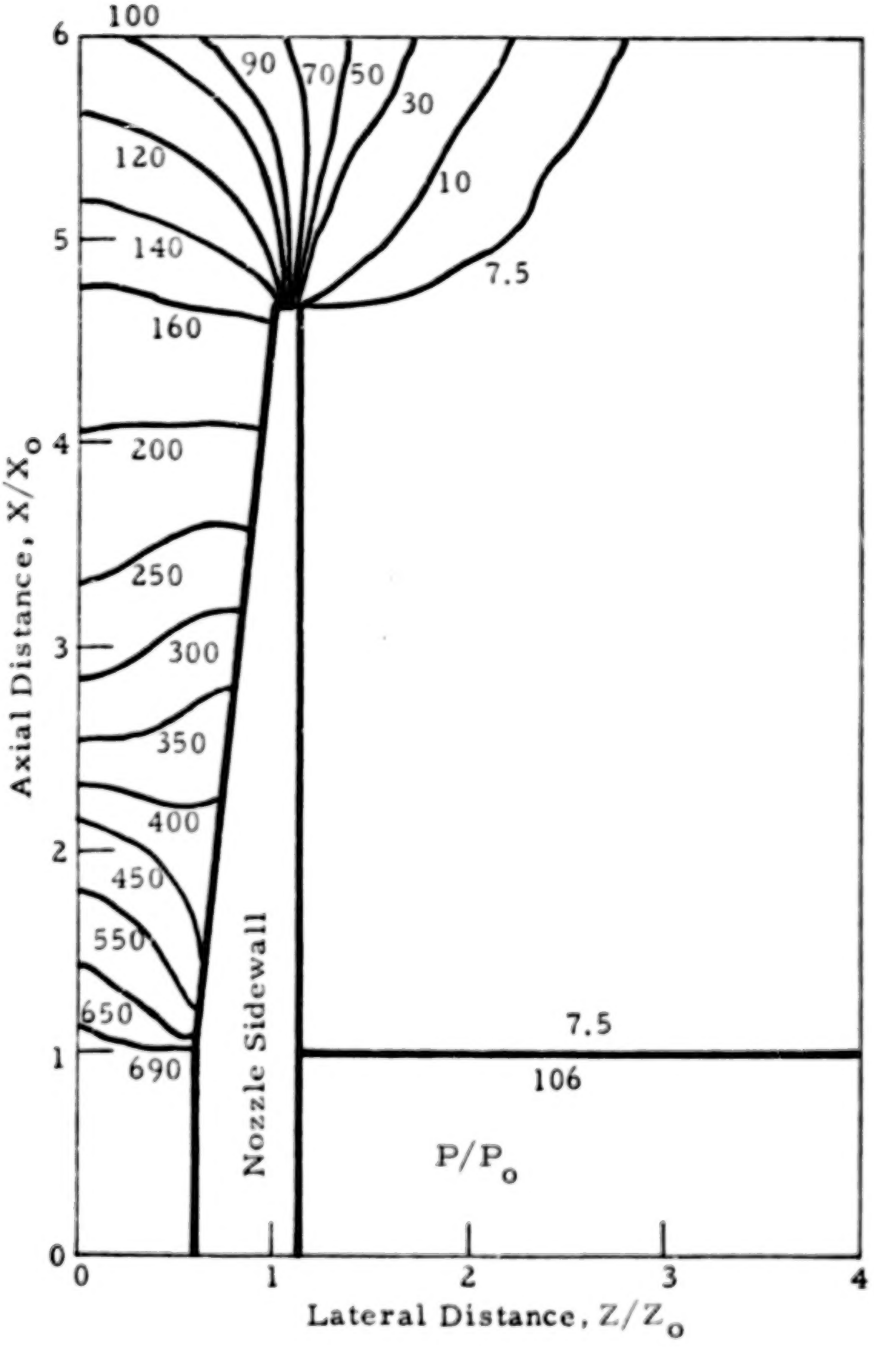


Fig. 3-27 - Pressure Contours on Upper Wall
($X_o = Y_o = Z_o = .3048$ m, $P_o = 47.87$ N/m^2)

4. CONCLUDING REMARKS

4.1 SUMMARY

The GIM code described in the preceding sections is in production status for two- and three-dimensional flows in arbitrary geometric domains. The capability of the production code is currently for laminar flows of nonreacting ideal gases. In addition, however, the GIM code is in a continuous development state with modifications and updating of capability. The following are among the advantages that GIM has over previous approaches.

- Geometric flexibility of finite element
- Integral conservation laws
- Knowledge base of finite difference
- Boundary conditions via quasi-variational
- Speed of finite difference
- No matrix inversion for explicit calculation
- Unlimited problem size
- Only methodology which leads to completely general fluid mechanics code.

There are classes of fluid dynamics problems which can be computed with much simpler techniques than GIM. However, the flexibility of the GIM algorithms make this code applicable to a wide variety of flows. Among the classes of problems for which GIM is especially well suited are the following:

- Flows with subsonic + supersonic regions
- Transient analyses
- Problems requiring full Navier-Stokes equations
- Complex Geometries
- Location of three-dimensional shock surfaces.

The following summary gives some of the potential areas of fluid dynamics in which the code could be applied, either in its current state or by further development.

4.2 GAS DYNAMICS

- The code can currently compute a variety of nozzle and exhaust plume flow fields for ideal gases in subsonic, transonic, and supersonic regimes.
- Solid rocket motor ballistics computations can be made for gas-only flow. Addition of solid particle equations would provide the capability of analyzing two phase transonic flow and internal gas dynamics for submerged nozzles.
- Addition of chemistry packages to the GIM code will allow computation of reacting gas dynamics for all types of rocket motor and plume flow fields.
- Use of the GIM code for analysis of gasdynamic laser devices could prove valuable for three-dimensional flows. This would require addition of gas kinetic calculations and a viscous mixing model. Routines for performing these calculations can be added from other present laser codes.
- The expandable equation sets which can be utilized by the GIM code allows inclusion of real viscous terms such that laminar or turbulent boundary layer analyses can be made.
- Inclusion of routines to compute laminar and/or turbulent viscosity coefficients would enable a variety of shear flows to be computed. Interaction of scramjet exhaust flow with freestream air can be treated, including a coupled treatment of the shear layer. Low-altitude rocket plume flows can be computed, including possible base recirculation with chemistry effects and mixing.

4.3 AERODYNAMICS

- The GIM code can currently compute subsonic, transonic and supersonic flows over arbitrary aerodynamic configurations. The code can be used for transonic airfoil design calculations although it cannot compete speed-wise with the existing "fast solvers." Extensions of the code will enable it to be used for complete airplane flowfield studies. Aircraft engine flows and interactions with the freestream could be computed for scramjet analyses, for example. Also, the potential exists for extending this methodology to a hyperbolic marching technique in supersonic flow regions.

- Aerodynamic simulations of wind tunnel tests could be made. Effects of tunnel walls and boundary layers can be included in computing flow over arbitrary shaped models of aircraft configurations. Addition of an appropriate set of boundary conditions and a viscosity model will be required.
- Missile aerodynamics presents another area of possible application of the GIM code. Flows over high angle of attack missiles require a three-dimensional analysis. Computation of flows with vortices may be possible by extension of the current code. Missile staging aerodynamics present another area of possible extension to the current GIM code.

5. REFERENCES

1. Roache, P. J., Computational Fluid Dynamics, Hermosa Publishers, Albuquerque, N. M., 1972.

2. Zienkiewicz, O. C., The Finite Element Method in Structural and Continuum Mechanics, McGraw-Hill, London, 1967.

3. Finlayson, B. A., The Method of Weighted Residuals and Variational Principles with Application in Fluid Mechanics, Heat and Mass Transfer, Academic Press, New York, 1972.

4. Prozan, R. J., L. W. Spradley, P. G. Anderson and M. L. Pearson, "The General Interpolants Method," AIAA Paper No. 77-642, Third Computational Fluid Dynamics Conference, Albuquerque, N. M., June 1977.

5. Gordon, W. J., and C. A. Hall, "Construction of Curvilinear Coordinate Systems and Applications to Mesh Generation," J. Numer. Math., Vol. 1, 1973, pp. 461-477.

6. MacCormack, R. W., "The Effect of Viscosity in Hypervelocity Impact Catering," AIAA Paper 69-354, Cincinnati, Ohio, May 1969.

7. Cuffel, R. F., L. H. Back and P. F. Massier, "Transonic Flow Fields in a Supersonic Nozzle with Small Throat Radius of Curvature," AIAA J., Vol. 7, July 1969.

8. Pond, J. E., I-S. Chang and M. R. Brashears, "Development of Interference Flowfield Calculation Method for Aerodynamic Heating of External Stores," LMSC-HREC TR D390486, Lockheed Missiles & Space Company, Huntsville, Ala., November 1974.

9. Dash, S. M., and P. D. Del Guidice, "Shock Capture Finite Difference and Characteristic Reference Plane Techniques for the Prediction of Three-Dimensional Nozzle-Exhaust Flow Fields," NASA CR-145366, NASA-Langley Research Center, Hampton, Va., 1978.

10. Salas, M. D., and J. L. Hunt of Langley Research Center, Private communication regarding the Langley SEAGULL Code, with L. W. Spradley of Lockheed-Huntsville, June 1977.

1 Report No. NASA CR-3042	2 Government Accession No.	3 Recipient's Catalog No.
4 Title and Subtitle Computation of Three-Dimensional Nozzle-Exhaust Flow Fields With the GIM Code		5 Report Date August 1978 6 Performing Organization Code
7 Author(s) L. W. Spradley, P. G. Anderson, M. L. Pearson		8 Performing Organization Report No. LMSC-HREC TB D477112-A 10 Work Unit No.
9 Performing Organization Name and Address Lockheed Missiles and Space Company, Inc. Huntsville Research and Engineering Center 4800 Bradford Drive Huntsville, AL		11 Contract or Grant No. Purchase Order L53289A 13 Type of Report and Period Covered Contractor Report
12 Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546		14 Sponsoring Agency Code

15 Supplementary Notes

Langley Technical Monitor: James L. Hunt
Final Report

16 Abstract

A new methodology is introduced for constructing numerical analogs of the partial differential equations of continuum mechanics. A general formulation is provided which permits classical finite element and many of the finite difference methods to be derived directly. The approach, termed the General Interpolants Method (GIM), is new in the sense that it can combine the best features of finite element and finite difference methods. A quasi-variational procedure is used to formulate the element equations, to introduce boundary conditions into the method and to provide a natural assembly sequence. Proceeding as in the Method of Weighted Residuals, it is shown that both finite element and finite difference analogs can be derived from the same point of departure. A derivation is given in terms of general interpolation functions from this procedure. Example computations for transonic and supersonic flows in two and three dimensions are given to illustrate the utility of GIM. A three-dimensional nozzle-exhaust flow field is solved including interaction with the freestream and a coupled treatment of the shear layer. Potential applications of the GIM code to a variety of computational fluid dynamics problems is then discussed in terms of existing capability or by extension of the methodology.

17 Key Words (Suggested by Author(s)) Computational Fluid Dynamics Nozzle-Exhaust Flows Three-Dimensional Flows	18 Distribution Statement Unclassified - Unlimited Subject Category 34

19 Security Classif. (of this report) Unclassified	20 Security Classif. (of this page) Unclassified	21 No. of Pages 73	22 Price* $5.25

* For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1978

70

END

DEC 5, 1978